

Année Universitaire : 2014/2015

HABILITATION A DIRIGER DES RECHERCHES

Contributions à la reconnaissance de formes structurelle et à l'analyse vidéo

Discipline : Informatique

présentée et soutenue publiquement

par

Donatello Conte

le 12/12/2014

JURY

| | | | |
|---------------------|---------|----------------------------|------------------------|
| Mme Isabelle | BLOCH | Professeur des universités | Telecom ParisTech |
| M Hubert | CARDOT | Professeur des universités | Université de Tours |
| M Jean-Michel | JOLION | Professeur des universités | INSA de Lyon |
| M Jean-Yves | RAMEL | Professeur des universités | Université de Tours |
| Mme Christine | SOLNON | Professeur des universités | INSA de Lyon |
| M Salvatore-Antoine | TABBONE | Professeur des universités | Université de Lorraine |

REMERCIEMENTS

JE tiens sincèrement à remercier les membres de mon jury d'avoir accepté d'évaluer ce mémoire et de participer à ma soutenance.

Je remercie Isabelle BLOCH d'avoir accepté d'être rapporteur de ce mémoire. Sa renommée internationale est la preuve de la qualité de ses recherches. Je lui suis reconnaissant pour sa disponibilité à entamer une collaboration avec moi.

Je remercie Jean-Michel JOLION d'avoir trouvé le temps, malgré toutes ses importantes occupations au Ministère, de participer à ma soutenance en tant que membre du jury. Depuis ma thèse, il m'a toujours soutenu et aidé dans de nombreuses occasions.

Merci à Christine SOLNON d'avoir accepté d'évaluer mon travail et de participer à ma soutenance. Je la remercie particulièrement pour son invitation en tant que chercheur invité dans le laboratoire LIRIS à Lyon : ce séjour fut pour moi un réel enrichissement professionnel, grâce auquel j'ai pu apprécier à la fois sa rigueur scientifique et sa convivialité.

Merci aussi à Antoine TABBONE qui me fait l'honneur de participer à mon jury. Bien que nous ne nous soyons rencontrés qu'une fois, j'estime la façon dont il m'a montré son appréciation de mon travail. J'espère que des collaborations futures pourront naître entre nous.

Je suis reconnaissant à Hubert CARDOT qui a accepté de rapporter sur ce manuscrit. Je le remercie également pour son accueil et son aide, professionnel et personnel, depuis mon aménagement à Tours.

Un grand merci à Jean-Yves RAMEL qui m'a beaucoup aidé, soutenu et conseillé dans ma démarche pour réussir à travailler en France et qui, encore et toujours, est d'une grande aide pour résoudre toutes mes questions scientifiques et d'autre nature, qui se posent quotidiennement. Il a également accepté d'être le référent de mon HDR et il m'a beaucoup conseillé dans la rédaction de ce mémoire. Pour toutes ces raisons, qu'il reçoive ici l'expression de ma profonde gratitude.

Je remercie tous les étudiants, doctorants et collègues, italiens et français, avec lesquels j'ai travaillé pendant toutes ces années, pour tout le travail réalisé. Un remerciement particulier va à Luc BRUN pour sa disponibilité et son amitié.

J'adresse également mes remerciements à tous mes collègues du Laboratoire et du Département d'Informatique de Tours pour l'accueil qu'ils m'ont accordé. Je pense particulièrement aux membres de l'équipe REAI qui me permettent de travailler dans une ambiance conviviale et dynamique.

A celle qui est toujours à mes côtés, à celle qui donne du sens à ma vie, à celle avec qui j'embrasse mon destin, et à celles qui sont ma source de joie, je voudrais dire mille fois merci.

Tours, le 14 novembre 2014.

TABLE DES MATIÈRES

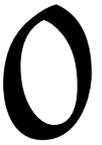
| | |
|---|------------|
| TABLE DES MATIÈRES | v |
| I CV et Activités scientifiques | vii |
| CV ET ACTIVITÉS SCIENTIFIQUES | ix |
| NOTICE INDIVIDUELLE | ix |
| FORMATION | ix |
| EMPLOI | x |
| ACTIVITÉS D'ENSEIGNEMENT | x |
| ACTIVITÉS DE RECHERCHE | xi |
| Parcours et thèmes de recherche | xi |
| Encadrements | xii |
| Animation de la recherche | xiii |
| Contrats et projets de recherche | xvii |
| ACTIVITÉS ADMINISTRATIVES ET COLLECTIVES | xviii |
| LISTE DE MES PUBLICATIONS | xx |
| 1 INTRODUCTION | 1 |
| II La Reconnaissance de Formes Structurelle | 3 |
| 2 NOTIONS DE BASE ET NOTATIONS | 5 |
| 2.1 DÉFINITIONS | 7 |
| 2.2 GRAPHERS ORIENTÉS ET NON ORIENTÉS | 8 |
| 2.3 SOUS-GRAPHERS | 9 |
| 2.4 CHEMINS, CIRCUITS ET CONNEXITÉ | 10 |
| 3 APPARIEMENT DE GRAPHERS | 13 |
| 3.1 MÉTHODES D'APPARIEMENT | 15 |
| 3.1.1 Définitions | 15 |
| 3.2 ANALYSE DE L'ÉTAT DE L'ART | 17 |
| 3.2.1 Algorithmes d'appariement basés sur les arbres de recherche | 17 |
| 3.2.2 Autres techniques | 20 |
| 3.2.3 Bilan de l'état de l'art | 21 |
| 3.3 NOS CONTRIBUTIONS À L'APPARIEMENT DE GRAPHERS | 22 |
| 3.3.1 L'évaluation des algorithmes | 22 |
| 3.3.2 L'appariement de pyramides de graphes | 26 |
| 3.4 PERSPECTIVES | 27 |

| | | |
|----------------------------|--|-----------|
| 4 | PLONGEMENT DE GRAPHERS | 31 |
| 4.1 | LE PLONGEMENT EXPLICITE : LE GRAPH EMBEDDING | 33 |
| 4.2 | LE PLONGEMENT IMPLICITE : LES NOYAUX DE GRAPHERS | 34 |
| 4.3 | CONTRIBUTIONS AU PLONGEMENT DE GRAPHERS | 35 |
| 4.3.1 | Des noyaux de graphes | 35 |
| 4.3.2 | Comparaison entre plongement implicite et explicite | 37 |
| 4.4 | PERSPECTIVES | 40 |
| III L'analyse vidéo | | 43 |
| 5 | INTRODUCTION AUX SYSTÈMES D'ANALYSE DE VIDÉOS | 45 |
| 5.1 | ARCHITECTURE D'UN SYSTÈME DE VIDÉO SURVEILLANCE | 47 |
| 5.2 | CONCLUSIONS | 49 |
| 6 | DÉTECTION D'OBJETS | 51 |
| 6.1 | ÉTAT DE L'ART | 53 |
| 6.2 | CONTRIBUTIONS | 55 |
| 6.3 | PERSPECTIVES | 73 |
| 7 | SUIVI D'OBJETS | 75 |
| 7.1 | ÉTAT DE L'ART | 77 |
| 7.2 | UNE CONTRIBUTION : SUIVI BASÉ SUR L'APPARIEMENT DE PYRAMIDES DE GRAPHERS | 79 |
| 7.3 | PERSPECTIVES | 84 |
| 7.4 | CONCLUSIONS | 88 |
| 8 | APPLICATIONS | 89 |
| 8.1 | ANALYSE DES TRAJECTOIRES | 91 |
| 8.2 | ESTIMATION DE FOULE | 93 |
| 8.3 | RÉ-IDENTIFICATION DE PERSONNES | 95 |
| 8.4 | PERSPECTIVES | 98 |
| CONCLUSION GÉNÉRALE | | 101 |
| BIBLIOGRAPHIE | | 103 |

Première partie

CV et Activités scientifiques

CV ET ACTIVITÉS SCIENTIFIQUES



NOTICE INDIVIDUELLE

CONTE Donatello
né le 09/11/1976 à Foggia(Italie)
marié, 2 enfants

Adresse professionnelle : Laboratoire d'Informatique
64, Avenue Jean Portalis
37200 Tours

N° de téléphone : +33 (0) 2 47 36 14 17

Adresse électronique : donatello.conte@univ-tours.fr

FORMATION

J'ai fait un doctorat en co-tutelle entre l'Université de Salerno (Italie) et l'INSA de Lyon.

Doctorat en France (Specialité Informatique) - juillet 2006

Intitulé : Detection, Tracking, and Behaviour Analysis of Moving People in Intelligent Video Surveillance Systems : A Graph Based Approach

Jury : L. Brun (Président), A. Deruyver (Rapporteur), A. Picariello (Rapporteur), J.M. Jolion (Directeur), M. Vento (Co-directeur)

Établissement ayant délivré le diplôme : INSA de Lyon

Organisme d'accueil : Laboratoire LIRIS

Doctorat en Italie (Ingénierie Informatique) - avril 2006

Intitulé : Detection, Tracking, and Behaviour Analysis of Moving People in Intelligent Video Surveillance Systems : A Graph Based Approach

Établissement ayant délivré le diplôme : Università di Salerno (Italie)

Organisme d'accueil : Laboratoire MIVIA

Diplôme d'ingénieur en Informatique - mars 2002

Intitulé : A Performance Analysis of three Maximum Common Subgraph Detection Algorithms

Établissement ayant délivré le diplôme : Università "Federico II" di Napoli (Italie)

EMPLOI

Je suis Maitre-de-Conférences depuis octobre 2006 en Italie, et en France depuis septembre 2013 suite à une mutation.

Situation professionnelle actuelle

Statut et fonction : Maitre de Conférences

Etablissement : Ecole Polytechnique de l'Université François Rabelais de Tours (<http://polytech.univ-tours.fr/>)

Date d'entrée en fonction : 1er septembre 2013

Laboratoire : Laboratoire d'Informatique (LI) de l'Université de Tours - Equipe RFAI (Reconnaissance de Formes et Analyse d'Images)

Situation professionnelle précédente

Statut et fonction : Maitre de Conférences

Etablissement : Università degli studi di Salerno (Italie) (www.unisa.it)

Date d'entrée en fonction : 1er octobre 2006

Laboratoire : MIVIA (Macchine Intelligenti per il riconoscimento di Video, Immagini e Audio) de l'Université de Salerno

ACTIVITÉS D'ENSEIGNEMENT

Le tableau ci-dessous récapitule le volume d'enseignement que j'ai effectués en tant que Maitre-de-Conférences en Italie de 2006 à 2012.

| Module | Niveau | Année | Volume |
|--|--------|-----------|-------------------------------|
| Langage C | L1 | 2008-2009 | 100h (30h CM, 40h TD, 30h TP) |
| | | 2009-2010 | 100h (30h CM, 40h TD, 30h TP) |
| | | 2010-2011 | 100h (30h CM, 40h TD, 30h TP) |
| | | 2011-2012 | 100h (30h CM, 40h TD, 30h TP) |
| | | 2012-2013 | 100h (30h CM, 40h TD, 30h TP) |
| Bases de données | L2 | 2006-2007 | 90h (30h CM, 30h TD, 30h TP) |
| | | 2007-2008 | 90h (30h CM, 30h TD, 30h TP) |
| | | 2008-2009 | 90h (30h CM, 30h TD, 30h TP) |
| | | 2009-2010 | 90h (30h CM, 30h TD, 30h TP) |
| Programmation d'applications distribuées | M2 | 2010-2011 | 50h (10h CM, 20h TD, 20h TP) |
| | | 2011-2012 | 50h (10h CM, 20h TD, 20h TP) |
| Programmation objet | L2 | 2007-2008 | 50h (10h CM, 20h TD, 20h TP) |
| Programmation pour le web | L3 | 2008-2009 | 50h (10h CM, 20h TD, 20h TP) |
| Architecture des ordinateurs | L2 | 2010-2011 | 50h (20h CM, 20h TD, 10h TP) |
| Méthodes pour la vision artificielle | M2 | 2010-2011 | 60h (20h CM, 20h TD, 20h TP) |

Le tableau ci-dessous récapitule le volume d'enseignement que j'ai effectués l'année dernière (2013-2014) à l'Ecole Polytechnique de l'Université de Tours.

| Module | Niveau (Bac+) | Volume h/eqTD |
|-------------------------|-------------------------|----------------------------|
| Traitement d'images | 3 (Diplôme d'ingénieur) | 6h (4h CM) |
| Administration Site Web | 4 (Diplôme d'ingénieur) | 48h (4h CM, 8h TD, 6h TP) |
| Technique Avancée Web | 5 (Diplôme d'ingénieur) | 28h (8h CM, 16h TP) |
| Imagerie médicale | 5 (Diplôme d'ingénieur) | 9h (6h CM) |
| Technologies sans fil | 5 (Diplôme d'ingénieur) | 54h (20h CM, 4h TD, 8h TP) |
| Internet & Réseau | 2 (Année Préparatoire) | 16h (8h TP) |

Outre les enseignements et les responsabilités administratives (que je détaillerai après), je participe également chaque année à l'encadrement de plusieurs projets de fin d'études d'élèves ingénieurs (dans la mesure de 3 encadrements par an).

Le volume horaire annuel de mes interventions est d'environ 220 heures équivalent TD.

ACTIVITÉS DE RECHERCHE

Cette partie décrit mon parcours et toutes les activités liées à la recherche.

Parcours et thèmes de recherche

Depuis mon début dans la recherche, j'ai effectué mes activités au sein d'un laboratoire en Italie et avec différents chercheurs français, ce qui m'a permis l'année dernière d'entrer dans l'université française avec mon recrutement à l'Université de Tours.

Mon parcours de recherche a débuté par un doctorat en co-tutelle sous la direction de Mario Vento de l'Université de Salerno et Jean-Michel Jolion de l'INSA de Lyon. Après l'obtention du diplôme de doctorat, j'ai rapidement intégré le laboratoire italien MIVIA en tant que Maître-de-Conférences. Jusqu'en 2013 j'ai principalement travaillé dans ce laboratoire, avec des nombreuses collaborations avec d'autres laboratoires français (LIRIS de Lyon, GreyC de Caen, LI de Tours).

Les collaborations ont concerné surtout la thématique de la Reconnaissance de Formes Structurelle et les graphes, grâce à la participation commune dans le Technical Committee 15 de l'IAPR. Le Technical Committee est l'un des comités techniques de l'association internationale de Reconnaissance de Formes. Il a pour but d'animer la recherche autour des graphes et des représentations structurelles des données.

Dans le laboratoire MIVIA, grâce à de nombreux projets industriels, j'ai pu développer mes compétences en analyse vidéo et en développement de systèmes de vidéo surveillance.

En conséquence, mes travaux peuvent se regrouper au sein de deux thématiques principales : la reconnaissance de formes structurelles et l'analyse vidéo.

Je décris ci-dessous, l'articulation chronologique de mon parcours et la manière dont ces deux thèmes ont été développés au cours des années 2002 à 2014.

2002-2006 - Laboratoire MIVIA (Université de Salerno)/ Laboratoire LIRIS (INSA de Lyon) - ma thèse, préparée sous la direction de Mario Vento en Italie et Jean-Michel Jolion en France, a porté sur le développement d'algorithmes de graph matching sur des pyramides de graphes. L'objectif était d'utiliser l'appariement de graphes pour résoudre le problème des occlusions dans le suivi d'objets.

2006-2013 - Laboratoire MIVIA (Université de Salerno) mon recrutement comme maître de conférences, en octobre 2006, au sein du Laboratoire MIVIA, m'a permis de renforcer les thématiques principales de ma recherche, à savoir la Reconnaissance de Formes avec les graphes et les systèmes de vidéo surveillance. Pendant cette période, j'ai eu nombreuses collaborations autour des noyaux de graphes avec Luc Brun du Laboratoire GreyC à Caen. Grâce à mes compétences en analyse vidéo, nous avons réalisé différents projets industriels qui ont permis la réalisation de systèmes pilotes de vidéo surveillance utilisés dans certaines gares de Rome. Ces activités de recherche ont abouti à la création, avec les collègues du Laboratoire MIVIA, d'une entreprise start-up qui développe des logiciels liés à l'analyse vidéo.

Début 2013 - Laboratoire LIRIS Au début de l'année 2013, j'ai passé trois mois dans la laboratoire LIRIS en tant que Professeur Invité. Pendant cette période, j'ai collaboré avec Guillaume Damiand et Christine Solnon pour traiter le problème des cartes généralisées, la définition de noyaux de cartes, et l'utilisation des cartes dans l'analyse vidéo. Pendant cette période, j'ai aussi eu une collaboration avec des chercheurs de plusieurs laboratoires (LI de Tours, GreyC de Caen, L3i de La Rochelle) autour du plongement de graphes.

Depuis 2013 - Laboratoire LI lors de ma mutation au laboratoire LI, j'ai intégré l'équipe RFAI, dans laquelle j'ai pu apporter mes compétences en analyse de vidéo avec ma participation à différents projets financés. J'ai également commencé à acquérir de nouvelles compétences en analyse de documents grâce à la participation à un projet industriel mené avec des collègues de l'équipe.

Encadrements

- Doctorants co-encadrés
 1. Oussama Hadjerci, Thèse dans le cadre du projet Région Centre DANIEAL (Laboratoire PRISME de Bourges et Laboratoire LI de Tours); *Titre* : Détection de Régions d'intérêt dans des Images Echographiques (en cours; première année); *Directeur de thèse* : Prof. P. Vieyres
 - Pourcentage de co-encadrement : 25%
 - A la fin de la première année, les premiers résultats du travail ont été publiés dans Hadjerci et al. (2014)
 2. Alessia Saggese, Thèse en co-tutelle entre Università di Salerno et ENSICAEN; *Titre* : Étude et développement de méthodes basées sur les noyaux sur graphes, pour des applications d'analyse vidéo dans le cadre de la vidéo surveillance (2010-2014; date de soutenance : 28.02.2014); *Co-directeurs de thèse* : Prof. L. Brun et Prof. M. Vento

- Pourcentage de co-encadrement : 30%
- Les résultats du travail de thèse ont été publiés dans Conte et al. (2012a)
- 3. Francesco Tufano, Università di Salerno ; *Titre* : Segmentazione di immagini biomediche con approcci ciechi ed approcci basati su modello (*Segmentation d'images médicales par des approches aveugles et des approches basées sur modèle*). (2005-2008 ; date de soutenance : 7.03.2008) ; *Directeur de thèse* : Prof. M. Vento
 - Pourcentage de co-encadrement : 50%
 - Les résultats du travail de thèse ont été publiés dans Conte et al. (2011), Conte et Tufano (2009), Conte et al. (2009)
- Post-Doctorants encadrés
 - Ali Reza Alaei (16 mois), Laboratoire LI de Tours, Equipe RFAI : activité de recherche dans le cadre du projet industriel "Compression d'images couleur" avec l'entreprise ITESoft

Animation de la recherche

Participation à des Conférences en tant que Président de Séance

1. The 8th Int. Conference on Computer Vision Theory and Applications (**VISAPP 2013**), Barcelona, Spain, 21-24 February, 2013
 - **Séance** : Image and Video Understanding - February, 22

Participation à des Comités de Programme

1. The 10th IAPR-TC15 Workshop on Graph-based Representations in Pattern Recognition. May 13 - 15, 2015, Beijing, China (**GbR 2015**).
2. The 10th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications. March 11-14, Berlin, Germany (**VISAPP 2015**).
3. 10th edition of the IEEE International Conference on Advanced Video and Signal-Based Surveillance. August 26-29, Seoul, Korea (**AVSS 2014**)
4. The sixth International Conference on Image and Signal Processing June 30-July 2, Cherbourg, Normandy, France (**ICISP 2014**).
5. The International Conference on Industrial Automation, Information and Communications Technology. 28 -30 August 2014, Bali, Indonesia (**IAICT 2014**).
6. The 2014 International Symposium on Control, Automation, Industrial Informatics and Smart Grid. 24-27 septembre, 2014. Delhi, India (**ICAIS 2014**).
7. The 9th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications. January 05-08, Lisbon, Portugal (**VISAPP 2014**).
8. World Congress on Multimedia and Computer Science. October 04-06, Hammamet, Tunisia. (**WCMCS 2013**).

9. The 3rd International Conference on Instrumentation, Control and Automation. August 28-30, Bali, Indonesia. (**ICA 2013**).
10. 2013 International Conference on Multi Media Processing. June 22-24, Sousse, Tunisia (**ICMMP 2013**)
11. The XXXIV International Conference on Information and Intelligent Systems. April 14-15, Venice, Italy (**ICIIS 2013**)
12. 10th edition of the IEEE International Conference on Advanced Video and Signal-Based Surveillance. August 27-30, Krakow, Poland (**AVSS 2013**)
13. The 2013 International Conference on Technology, Informatics, Management, Engineering & Environment. June 23-26, 2013. Bandung, Indonesia (**TIME-E 2013**)
14. The 2013 IEEE Conference on Control, Systems & Industrial Informatics. June 23-26, 2013. Bandung, Indonesia (**ICCSII 2013**)
15. The 2013 International Symposium on Control, Automation, Industrial Informatics and Smart Grid. August 24-25, 2013. Mysore, India (**ICAIS 2013**)
16. 9th IAPR - TC15 Workshop on Graph-based Representations in Pattern Recognition. May 15-17, 2013. Vienna, Austria. (**GbR 2013**)
17. The 2013 IEEE Workshop on the Applications of Computer Vision. January 17-18, 2013. Clearwater Beach, Florida. (**WACV 2013**)
18. The 2012 IEEE Student Conference on Research and Development. December 5-6, 2012. Penang, Malaysia (**SCORED 2012**)
19. The 2012 IEEE Conference on Control, Systems & Industrial Informatics. September 23-26, 2012. Bandung, Indonesia (**ICCSII 2012**)

Activités de Relecture d'articles

Relectures pour des *revues internationales* :

1. IEEE Transactions on Pattern Analysis and Machine Intelligence
2. Communications in Statistics - Simulation and Computation
3. The Visual Computer
4. IEEE Transactions on Neural Networks and Learning Systems
5. British Journal of Mathematics & Computer Science
6. AI Communications
7. Annals of Operations Research
8. British Journal of Mathematics & Computer Science
9. Electronic Letters on Computer Vision and Image Analysis
10. IEEE Transactions on Neural Networks
11. Imaging and Vision Computing
12. International Journal of Cooperative Information Systems
13. Journal of Graph Algorithms and Applications

14. Machine Vision and Applications
15. Pattern Recognition
16. Pattern Recognition Letters
17. The Visual Computer

Relectures pour des *conférences internationales* :

1. The 10th IAPR-TC15 Workshop on Graph-based Representations in Pattern Recognition. May 13-15, 2015, Beijing, China (**GbR 2015**).
2. The 10th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications. March 11-14, Berlin, Germany (**VISAPP 2015**).
3. The XXVII Conference on Graphics, Patterns and Images. August 27-30, Rio de Janeiro, Brazil (**SIBGRAPI 2014**)
4. 10th edition of the IEEE International Conference on Advanced Video and Signal-Based Surveillance. August 26-29, Seoul, Korea (**AVSS 2014**)
5. The sixth International Conference on Image and Signal Processing June 30-July 2, Cherbourg, Normandy, France (**ICISP 2014**).
6. The International Conference on Industrial Automation, Information and Communications Technology. 28-30 August 2014, Bali, Indonesia (**IAICT 2014**).
7. The 2014 International Symposium on Control, Automation, Industrial Informatics and Smart Grid. 24-27 septembre, 2014. Delhi, India (**ICAIS 2014**).
8. The 9th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications. January 05-08, Lisbon, Portugal (**VISAPP 2014**).
9. The 2013 International Conference on Robotics, Biomimetics, Intelligent Computational Systems. 25-27 November, Jogjakarta, Indonesia (**ROBIONETICS 2013**).
10. World Congress on Multimedia and Computer Science. October 04-06, Hammamet, Tunisia. (**WCMCS 2013**).
11. The 3rd International Conference on Instrumentation, Control and Automation. August 28-30, Bali, Indonesia. (**ICA 2013**).
12. 2013 International Conference on Multi Media Processing. June 22-24, Sousse, Tunisia (**ICMMP 2013**)
13. The XXXIV International Conference on Information and Intelligent Systems. April 14-15, Venice, Italy (**ICIIS 2013**)
14. 10th edition of the IEEE International Conference on Advanced Video and Signal-Based Surveillance. August 27-30, Krakow, Poland (**AVSS 2013**)
15. The 2013 International Conference on Technology, Informatics, Management, Engineering & Environment. June 23-26, 2013. Bandung, Indonesia (**TIME-E 2013**)

16. The 2013 IEEE Conference on Control, Systems & Industrial Informatics. June 23-26, 2013. Bandung, Indonesia (**ICCSII 2013**)
17. The 2013 International Symposium on Control, Automation, Industrial Informatics and Smart Grid. August 24-25, 2013. Mysore, India (**ICAIS 2013**)
18. The 2013 IEEE Workshop on the Applications of Computer Vision. January 17-18, 2013. Clearwater Beach, Florida. (**WACV 2013**)
19. The 2012 IEEE Student Conference on Research and Development. December 5-6, 2012. Penang, Malaysia (**SCORED 2012**)
20. The 2012 IEEE Conference on Control, Systems & Industrial Informatics. September 23-26, 2012. Bandung, Indonesia (**ICCSII 2012**)
21. The 9th International Conference on Advanced Video and Signal-Based Surveillance. Septembre 18-21, 2012. Beijing, China (**AVSS 2012**)
22. The 3rd International Conference on Advanced Cognitive Technologies and Applications. September 25-30, 2011 - Rome, Italy (**COGNITIVE 2011**)
23. The 8th IAPR - TC-15 Workshop on Graph-based Representations in Pattern Recognition May 18-20, 2011 Münster (Germany) (**GbR 2011**).
24. The 13th International Conference on Computer Analysis of Images and Patterns. Münster (Germany). September 2-4, 2009 (**CAIP 2009**).
25. 15th International Conference on Image Analysis and Processing. Vietri sul mare (Italy). September 8 - 11, 2009 (**ICIAP 2009**).
26. 18th International Conference on Pattern Recognition. Hong Kong. August 20 - 24, 2006 (**ICPR 2006**).
27. IEEE Intern. Workshop on Computer Architecture for Machine Perception (**CAMP05**) Terrasini - Palermo, 4-6 July, 2005
28. Joint IAPR International Workshops on Syntactical and Structural Pattern Recognition (**SSPR 2004**) and Statistical Pattern Recognition (SPR 2004). Lisbon, Portugal, August 18 - 20, 2004.
29. 17th International Conference on Pattern Recognition. Cambridge, UK. August 23 - 26, 2004 (**ICPR 2004**).

Autres activités liées à la recherche

- 2013 - présent : Expert scientifique au titre du crédit impôt-recherche
- 2002 - présent : Membre du Technical Committee #15 du IAPR, dédié à la promotion de la recherche dans le domaine de la reconnaissance des formes basée graphes (Graph-based Representations in the Pattern Recognition).
- 2013 - présent : Membre de l'Association Française pour la Reconnaissance et l'Interprétation des Formes (AFRIF)
- 2002 - présent : Membre de l'Association IAPR International Association for Pattern Recognition (IAPR)
- 2002 - 2014 : Membre du Groupe Italien des Chercheurs en Pattern Recognition (Group of Italian Researcher in Pattern Recognition - GIRPR)

- 2009 : Membre du Comité Local de la conférence internationale ICIAP 2009 : 15th International Conference on Image Analysis and Processing, Vietri Sul Mare (SA), Italy, September 8-11, 2009.

Contrats et projets de recherche

- 2014-2015 : Co-responsable du projet industriel "Compression d'images couleur" avec l'entreprise ITESoft, SA. Dans le projet il y a deux WP : je suis responsable du WP1 : - "Optimisation du taux de compression par l'adaptation dynamique des paramètres de compression en fonction de la qualité de l'image" ; Budget total du projet : 180.000 € ; *Encadrement d'un PostDoc* ;
- 2013-2015 : Membre de l'équipe de recherche du projet financé par la Région Centre : "DANIEAL : Détection et Analyse des Nerfs dans les Images Echographiques pour l'Anesthésie Locorégionale" ; Budget total du projet : 444.000 € dont 198.000 € financés par la Région ; *Encadrement d'un doctorant* ;
- 2013-2015 : Membre de l'équipe de recherche du projet financé par la Région Centre : "OVIN2A : Identification des causes de mortalité des agneaux élevés en allaitement artificiel" ; Budget total du projet : 706.000 € dont 198.000 € financés par la Région ; *Encadrement de projets de fin d'études sur le développement de logiciels d'analyse des comportements par le biais du traitement de vidéos* ;
- 2012 : Collaboration avec l'entreprise Rubrik srl (Italie) sur le projet : "Design a document management system" ;
- 2009-2010 : Collaboration avec l'entreprise SESM scarl (Finmeccanica Group - Italie) sur le projet : "Design and development of an anomaly detection system for airport security" ;
- 2006-2010 : Collaboration avec l'entreprise Nexera srl (Italie) sur la réalisation d'un système de vidéo surveillance intelligent pour la sécurités de certaines gares de Naples et Rome ;
- 2006-2010 : Membre de l'équipe de recherche du projet national (Italie) : "ESALAB - Development of innovative technologies for the advanced medical diagnostics" - Responsable du projet : Prof. S. Russo (Université de Naples - Italie)
Budget total du projet : 1.431.698 € ;
- 2004-2006 : Membre de l'équipe de recherche des projets régionaux (Italie) :
 - CERBERO : A multimodal and multichannel access control system
Budget total du projet : 491.230 € ;

- ANGELO : Systems for providing innovative value-added services for the citizen movements and multimedia in the territory
Budget total du projet : 712.000 €;
- PROTEO : An advanced e-Procurement system
Budget total du projet : 500.750 €;
- 2003-2005 : Membre de l'équipe de recherche du projet national (Italie) : "Un sistema per l'analisi computerizzata e la fruizione in rete di immagini mammografiche per la diagnosi precoce del cancro alla mammella." (Un système informatique pour l'analyse des images mammographiques pour le diagnostic précoce du cancer du sein) - Responsable du projet : Prof. M. Vento
Budget total du projet : 147.700 €

Transfert de technologie et création d'entreprise

En 2010, j'ai participé à la création d'une entreprise, start-up académique, dénommée "AI Tech srl" (www.aitech-solutions.eu) avec les autres membres du laboratoire MIVIA.

L'entreprise commercialise de nouveaux produits dans les domaines suivants :

- Surveillance intelligente du trafic à travers l'analyse de la vidéo en temps réel ;
- Analyse et interprétation du son pour des applications de sécurité ;
- Analyse et interprétation d'images médicales.

A.I. Tech srl a développé des partenariats stratégiques avec des entreprises leaders dans la production de caméras (Axis - <http://www.axis.com/>) et dans le domaine de la vidéo surveillance (Milestone Systems - <http://www.milestonesys.com/>).

ACTIVITÉS ADMINISTRATIVES ET COLLECTIVES

Dès ma prise de fonction en tant que Maitre-de-Conférences en Italie, puis en France, j'ai toujours assumé certaines responsabilités administratives pour mes départements d'appartenance.

Dans le département dans lequel je travaille actuellement :

- Responsable pédagogique de la quatrième année du Département d'informatique (Polytech Tours) : Gestion des emplois du temps et des intervenants, suivi pédagogique, relations avec les élèves, ...
- Responsable du thème "Système d'Information, Réseau et Web" du Diplôme d'ingénieur du Département d'informatique : gestion des intervenants, réunions pédagogiques, ...

Dans le Département d'Ingénierie de l'Information et des Mathématiques Appliquées en Italie :

- Responsable des projets de stages des étudiants de licence (contacts avec les entreprises, suivi des projets, etc.)

- Responsable de la Commission de suivi statistique (la commission fait des rapports statistiques sur les activités pédagogiques du département)
- Membre de la Commission de Gestion de la Qualité (la commission s'occupe des activités de planification de la qualité, contrôle qualité, assurance qualité et amélioration de la qualité de l'enseignement).

LISTE DE MES PUBLICATIONS

Revue internationale

1. D. Conte, P. Foggia, G. Percannella et M. Vento. "Counting moving persons in crowded scenes". In : *Machine Vision and Applications* 24 - 5 (2013), p. 1029-1042. doi : 10.1007/s00138-013-0491-3.
2. D. Conte, P. Foggia, G. Percannella et M. Vento. "A 3D Appearance Model for Object Tracking in Video Surveillance Applications". In : *The Mediterranean Journal of Computers and Networks* 8-4 (2012), p. 134-144.
3. D. Conte, P. Foggia, G. Percannella et M. Vento. "Removing object reflections in videos by global optimization". In : *IEEE Transactions on Circuits and Systems for Video Technology* 22-11 (2012), p. 1623-1633.
4. D. Conte, P. Foggia, G. Percannella, F. Tufano et M. Vento. "A Method for Counting Moving People in Video Surveillance Videos". In : *EURASIP Journal on Advances in Signal Processing* 2010 :231240 (2010), p. 10.
5. D. Conte, P. Foggia, G. Percannella, F. Tufano et M. Vento. "An Experimental Evaluation of Foreground Detection Algorithms in Real Scenes". In : *EURASIP Journal on Advances in Signal Processing* 2010 :373941 (2010), p. 11.
6. D. Conte, P. Foggia et M. Vento. "Challenging Complexity of Maximum Common Subgraph Detection Algorithms : A Performance Analysis of Three Algorithms on a Wide Database of Graphs". In : *Journal of Graph Algorithms and Applications* 11-1 (2007), p. 99-143.
7. D. Conte, P. Foggia, J.M. Jolion et M. Vento. "A Graph-based, Multi-resolution Algorithm for Tracking Objects in Presence of Occlusions". In : *Pattern Recognition Journal* 39-4 (2006), p. 562-572.
8. D. Conte, P. Foggia, C. Sansone et M. Vento. "Thirty years of Graph Matching in Pattern Recognition". In : *International Journal of Pattern Recognition and Artificial Intelligence* 18-3 (2004), p. 265-298.

Chapitres de Livres

1. D. Conte, P. Foggia, F. Tufano et M. Vento. "An Enhanced Level Set Algorithm for Wrist Bone Segmentation". In : *Image Segmentation*. Sous la dir. de Pei-Gee Ho (Ed.) InTech, 2011, p. 293-308.
2. D. Conte, P. Foggia, C. Sansone et M. Vento. "How and Why Pattern Recognition and Computer Vision Applications Use Graphs". In : *Applied Graph Theory in Computer Vision and Pattern Recognition*. Sous la dir. de A. Kandel, H. Bunke et M. Last (Eds.) Studies in Computational Intelligence. Springer-Verlag, 2007, p. 85-135.

Conférences internationales avec comité de lecture et publication des actes

1. D. Conte et G. Damiand. "Remove noise in video with 3D topological maps". In : *Workshops on Statistical Techniques in Pattern Recognition*

- tion and Structural and Syntactic Pattern Recognition (SSPR), LNCS 8621. 2014.
2. O. Hadjerci, A. Hafiane, P. Makris, D. Conte, P. Vieyres et A. Delbos, "Nerve detection in ultrasound images using median Gabor binary pattern". In : Proceedings of the International Conference on Image Analysis and Recognition (ICIAR). october 2014.
 3. A. Mahboubi, L. Brun, D. Conte, P. Foggia et M. Vento. "Tracking System with Re-identification using a RGB String Kernel". In : Proceedings of the IAPR International Workshops on 15th Structural and Syntactic Pattern Recognition Workshop (SSPR), LNCS 8621. 2014.
 4. D. Conte, R. Di Lascio, P. Foggia, G. Percannella et M. Vento. "Pupil Localization by a Template Matching Method". In : Proceedings of the International Conference on Computer Vision Theory and Applications (VISAPP 2013). T. 2. 2013, p. 779-782.
 5. D. Conte, J.-Y. Ramel, N. Sidère, M.M. Luqman, B. Gauzère, J. Gibert, L. Brun, et M. Vento. "A Comparison of Explicit and Implicit Graph Embedding Methods for Pattern Recognition". In : Proceedings of the 9th IAPR-TC15 workshop on Graph-based Representation in Pattern Recognition (GbR 2013). 2013, p. 81-90.
 6. A. Mahboubi, L. Brun, D. Conte, P. Foggia et M. Vento. "Tracking System with Reidentification Using a Graph Kernels Approach". In : Proceedings of the 15th International Conference on Computer Analysis of Images and Patterns. 2013, p. 401-408.
 7. D. Conte, P. Foggia, G. Percannella, A. Saggese et M. Vento. "An Ensemble of Rejecting Classifiers for Anomaly Detection of Audio Events". In : Proceedings of the 9th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS 2012). 2012, p. 76-81.
 8. L. Brun, D. Conte, P. Foggia et Vento M. "A Graph-Kernel Method for Reidentification". In : Proceedings of the 8th International Conference on Image Analysis and Recognition (ICIAR 2011). Sous la dir. de M. Kamel et A. Campilho (Eds.) T. 6753. Lecture Notes in Computer Science. 2011, p. 173-182.
 9. L. Brun, D. Conte, P. Foggia et Vento M. "People Re-identification by Graph Kernels Methods". In : Proceedings of the 8th IAPR-TC-15 International Workshop, GbRPR 2011. Sous la dir. de Xiaoyi Jiang, Miquel Ferrer et Andrea Torsello (Eds.) T. 6658. Lecture Notes in Computer Science. 2011, p. 285-294.
 10. D. Conte, P. Foggia, G. Percannella, F. Tufano et M. Vento. "An Effective Method For Counting People In Video-Surveillance Applications". In : Proceedings of the 6th Conference on Computer Vision Theory and Applications (VISAPP 2011). 2011.
 11. D. Conte, P. Foggia, G. Percannella, F. Tufano et M. Vento. "Reflection Removal for People Detection in Video Surveillance Applications". In : Proceedings of the 16th International Conference on Image Analysis and Processing (ICIAP 2011). T. 6978. Lecture Notes in Computer Science. 2011, p. 178-186.

12. D. Conte, P. Foggia, G. Percannella et M. Vento. "Long-term Object Tracking by MultiView Appearance Model". In : Proceedings of the 5th Conference on Conference on Software, Knowledge Information, Industrial Management and Applications (SKIMA 2011). 2011.
13. L. Brun, D. Conte, P. Foggia, M. Vento et D. Villemin. "Symbolic Learning vs. Graph Kernels : An Experimental Comparison in a Chemical Application". In : Proceedings of the 14th Conference on Advances in DataBases and Information Systems (ADBIS 2010). 2010, p. 31-40.
14. D. Conte, P. Foggia, G. Percannella, F. Tufano et M. Vento. "A Method for Counting People in Crowded Scenes". In : Seventh IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS 2010). 2010, p. 225-232.
15. D. Conte, P. Foggia, G. Percannella, F. Tufano et M. Vento. "An algorithm for recovering camouflage errors on moving people". In : Joint IAPR International Workshop on Structural, Syntactic, and Statistical Pattern Recognition. Sous la dir. d'Edwin R. Hancock, Richard C. Wilson, Terry Windeatt, Ilkay Ulusoy et Francisco Escolano (Eds.) T. 6218. Lecture Notes in Computer Science. 2010, p. 365-374.
16. D. Conte, P. Foggia, G. Percannella, F. Tufano et M. Vento. "Counting Moving People in Videos by Salient Points Detection". In : 20th International Conference on Pattern Recognition (ICPR2010). 2010, p. 1743-1746.
17. D. Conte, P. Foggia, G. Percannella, F. Tufano et M. Vento. "Reflection removal in color videos". In : 20th International Conference on Pattern Recognition (ICPR 2010). 2010, p. 1788-1791.
18. D. Conte, P. Foggia, G. Percannella et M. Vento. "A Method Based on the Indirect Approach for Counting People in Crowded Scenes". In : Seventh IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS 2010). 2010, p. 111-118.
19. D. Conte, P. Foggia, G. Percannella et M. Vento. "Performance Evaluation of a People Tracking System on PETS2009 Database". In : Seventh IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS 2010). 2010, p. 119-126.
20. D. Conte, P. Foggia, G. Percannella, F. Tufano et M. Vento. "An algorithm for detection of partially camouflaged people". In : Proceedings of the IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS 2009). 2009, p. 340-345.
21. D. Conte, P. Foggia, F. Tufano et M. Vento. "Evaluation And Improvements Of The Level Set Method For RM Images Segmentation". In : Proceedings of International Conference on Computer Vision Theory and Applications (VISAPP 2009). 2009, p. 210-215.
22. D. Conte et F. Tufano. "Segmenting Wrist Bone Tissue : Enhancements to Level Set Algorithm". In : Proceedings of the International Workshop on Medical Measurements and Applications (MeMeA 2009). 2009, p. 177-182.

23. D. Conte, P. Foggia, J. M. Jolion et M. Vento. "A Graph-based, Multi-resolution Algorithm for Tracking Objects in Presence of Occlusions". In : Proceedings of the 5th Workshop on Graph-based Representation in Pattern Recognition (GbRPR2005). Sous la dir. de L. Brun et M. Vento (Eds.) T. 3434. Lecture Notes in Computer Science. 2005, p. 193-202.
24. D. Conte, P. Foggia, M. Petretta, F. Tufano et M. Vento. "Evaluation and Improvements of a Real-Time Background Subtraction Method". In : Proceedings of the International Conference on Image Analysis and Recognition (ICIAR 2005). Sous la dir. de M. Kamel et A. Campilho (Eds.) T. 3656. Lecture Notes in Computer Science. 2005, p. 1234-1241.
25. D. Conte, P. Foggia, M. Petretta, F. Tufano et M. Vento. "Meeting the application requirements of intelligent video surveillance systems in moving object detection". In : Proceedings of the International Conference on Advances in Pattern Recognition (ICAPR 2005). Sous la dir. de S. Singh, M. Singh, C. Apte et P. Perner (Eds.) T. 3687. Lecture Notes in Computer Science. 2005, p. 653-662.
26. D. Conte, P. Foggia, C. Guidobaldi, A. Limongiello et M. Vento. "An Object Tracking Algorithm Combining Different Cost Functions". In : Proceedings of the International Conference on Image Analysis and Recognition (ICIAR 2004). T. 3212. Lecture Notes in Computer Science. 2004, p. 614-622.
27. D. Conte, P. Foggia, C. Sansone et M. Vento. "Graph Matching Applications in Pattern Recognition and Image Processing". In : Proceedings of the IEEE International Conference on Image Processing (ICIP2003). T. II. 2003, p. 21-24.
28. D. Conte, C. Guidobaldi et C. Sansone. "A Comparison of Three Maximum Common Subgraph Algorithms on a Large Database of Labeled Graphs". In : Proceedings of the International Workshop on Graph-based Representation in Pattern Recognition (GbR 2003). Sous la dir. de E. Hancock et M. Vento (Eds.) T. 2726. Lecture Notes in Computer Science. 2003, p. 130-141.

Conférences Nationales avec comité de lecture et publication des actes

1. A. Mahboubi, L. Brun, D. Conte, P. Foggia et M. Vento. "Ré-identification de Personnes par Modèle de Noyaux de Graphe". In : 24ème Edition du colloque GRETSI. 2013.
2. D. Conte, P. Foggia, G. Percannella, F. Tufano et M. Vento. "A Method for Counting People in Crowded Scenes". In : Atti del V Convegno del Gruppo Italiano Ricercatori in Pattern Recognition. 2010.
3. D. Conte, P. Foggia, J.M. Jolion et M. Vento. "Un algorithme multi-résolution pour la gestion des occlusions basé sur les pyramides de graphes". In : Dixième Edition de la conférence sur COMpression et REprésentation des Signaux Audiovisuel (CORESA 2005). 2005.

4. D. Conte et al. "Object detection and tracking for video surveillance". In : Atti del II Convegno del Gruppo Italiano Ricercatori in Pattern Recognition. 2004.
5. D. Conte et al. "Acquisizione e Fruizione su Internet di Opere d'arte". In : Atti del Convegno Contesti virtuali e fruizione dei beni culturali. 2003.

INTRODUCTION



Reprenant la définition de C. Bishop [Bishop (2006)], la Reconnaissance de Formes (RF) concerne la découverte automatique de régularités dans les données qui permettent d'effectuer des actions telles que la classification de données dans différentes catégories et la prise de décision suite à cette classification.

On peut faire remonter cette discipline au temps des premiers ordinateurs et des premiers processus algorithmiques. Ses bases mathématiques remontent même au XVIII^e siècle (avec Bayes, Laplace, Euler, etc.).

C'est pourtant à partir des années 1960 que l'on peut situer les premières méthodologies de RF stables et reconnues telles que l'Analyse par Composantes Principales (ACP), les Réseaux de Neurones, et vers les années 1980 pour retrouver les techniques basées sur la probabilité Bayésienne (les méthodes Monte Carlo, les chaînes de Markov, etc.). Toutes les méthodes que nous venons de citer se basent sur une représentation numérique des données par des vecteurs ou matrices et elles font partie de la branche de la RF appelée RF statistique.

L'utilisation d'autres structures de données, différentes des vecteurs et des matrices, est plus récente. L'emploi de données structurées dans la RF, a fait naître la branche de la RF structurelle. Parmi les structures de données utilisées, celle qui a la place la plus importante en nombre de contributions, de conférences et de numéros spéciaux de revues internationales est la représentation à base de graphes. Leur popularité croissante dans les domaines de la RF et de la vision par ordinateur peut être expliquée par la capacité des graphes à représenter des formes complexes à travers leur capacité à modéliser des interactions entre composantes plus simples. Cela constitue une évidente différence par rapport aux plus conventionnelles représentations vectorielles. La représentation par graphe présente dans certains cas un nombre de propriétés invariantes (comme la rotation, la translation, etc.) qui pourraient être très intéressantes dans différents problèmes.

Cependant, nous avons à payer un prix lors de l'usage de ce type de représentations enrichies et intéressantes : la complexité computationnelle des opérations. Cela rend la RF structurelle encore très ouverte à des nouvelles contributions et d'intéressantes découvertes restent donc à réaliser.

Mon travail de recherche se situe dans ce contexte et ce mémoire retrace le parcours scientifique que j'ai suivi depuis mes premières activités de recherche commencées avec mes études doctorales. Son fil conducteur est l'étude et la conception de nouveaux algorithmes basés sur l'utilisation de données structurelles dans la RF. Au cours de mes recherches, j'ai voulu

d'abord produire une étude et une catégorisation systématique d'une majorité des travaux de recherche du domaine pour bien comprendre le contexte et pour bien situer mes nouvelles propositions. Ce travail, puisqu'il comblait une lacune présente dans la communauté de la RF structurelle, a été très apprécié et m'a permis d'entrer rapidement en contact avec les acteurs principaux de cette discipline, avec qui j'ai eu l'opportunité de monter d'intéressantes collaborations scientifiques qui ont porté leurs fruits.

Le cadre d'application privilégié de mes travaux de recherche grâce à de nombreux projets industriels auxquels j'ai participé, a été l'analyse vidéo et en particulier la vidéo surveillance. De ce fait, dans mes propositions j'ai toujours eu une attention particulière concernant la fiabilité et l'efficacité des solutions proposées dans un cadre réel d'applications. Mes compétences en développement de systèmes concrets ont même abouti à la création récente, avec d'autres collègues, d'une start-up académique, qui produit des logiciels autour du domaine de la vidéo surveillance et de l'analyse vidéo. Pour cette raison une deuxième partie de ce mémoire traite le domaine de l'analyse de vidéos.

Le plan de ce mémoire est le suivant. Il est divisé en deux parties : une première partie est dédiée à la RF structurelle et l'autre partie à l'analyse de vidéo. Nous présentons, au chapitre 2, les notions de base et les notations utilisées sur les graphes qui seront utiles dans la suite de ce mémoire. Le chapitre 3 introduit les problèmes d'appariement de graphes, outils parmi les plus utilisés dans la RF structurelle. Dans le chapitre 4, on s'est intéressé à l'une des plus prometteuses techniques récentes d'utilisation de l'information structurelle dans la RF : le plongement de graphes. La deuxième partie du mémoire concerne l'analyse de vidéos et plus particulièrement des applications de vidéo surveillance. Après avoir décrit l'architecture générale d'un système de vidéo surveillance (chapitre 5) nous décrirons les différentes composantes dans les chapitres suivants : la détection des objets (chapitre 6), le suivi d'objets (chapitre 7) et les différentes applications dans lesquelles nous avons exploité nos algorithmes (chapitre 8).

Enfin nous présentons une conclusion générale ainsi que quelques perspectives futures à la fin du manuscrit.

Deuxième partie

**La Reconnaissance de Formes
Structurelle**

NOTIONS DE BASE ET NOTATIONS

2

SOMMAIRE

| | |
|--|----|
| 2.1 DÉFINITIONS | 7 |
| 2.2 GRAPHES ORIENTÉS ET NON ORIENTÉS | 8 |
| 2.3 SOUS-GRAPHES | 9 |
| 2.4 CHEMINS, CIRCUITS ET CONNEXITÉ | 10 |

LES graphes sont des structures de données puissantes largement utilisées dans de nombreux domaines scientifiques. Leur succès est principalement lié à la simplicité qu'ils engendrent pour représenter les entités par des sommets et les relations existantes entre ceux-ci.

L'objectif de ce chapitre est de présenter quelques notions de base sur la théorie des graphes qui seront utiles dans la suite de ce mémoire. Ce chapitre nous sera aussi utile pour illustrer les notations utilisées dans le reste du document.

Ce chapitre est organisé de la manière suivante. Nous présentons dans la Section 2.1 la définition formelle, ainsi que quelques exemples, du concept de graphe. Nous introduisons ensuite Section 2.2 les deux principales variantes de graphes, c'est-à-dire les graphes orientés et les graphes non orientés. Deux autres notions importantes pour bien appréhender la suite de ce mémoire, à savoir les sous-graphes et les chemins et circuits dans un graphe, seront illustrées dans la Section 2.3 et la Section 2.4.

Les graphes sont couramment utilisés en recherche opérationnelle : le problème du voyageur de commerce, le problème du plus court chemin sont deux des exemples les plus connus de problèmes sur graphe. Dans ce cadre, il existe des algorithmes très connus qui trouvent la solution à certains de ces problèmes en temps polynomial. Les problèmes cités tombent dans la catégorie des problèmes d'optimisation sur graphe. Dans ces problèmes, un seul graphe représente l'ensemble des données considérées et les algorithmes opèrent donc sur un unique graphe, tandis que dans la RF un graphe représente généralement un modèle parmi d'autres, ou un objet dans un ensemble d'objets à classifier ; donc dans la RF, on traite plutôt des problèmes de comparaison entre graphes. Dans ce chapitre, on ne rappelle que les définitions sur les graphes utiles dans le contexte de la RF.

2.1 DÉFINITIONS

La définition formelle d'un graphe est la suivante.

Définition 2.1.1 (Graphe) *Un graphe G est une quadruple $G(V, E, \nu, \zeta)$, où :*

- V est un ensemble $\{v_1, v_2, \dots, v_n\}$ de sommets,
- E est un ensemble $\{e_k = (v_i, v_j) : v_i \text{ et } v_j \in V \text{ avec } k = 1, \dots, m\}$ d'arêtes,
- ν est une fonction

$$\nu : v \in V \rightarrow \nu(v)$$

qui associe à chaque sommet un attribut qui peut être simple (une valeur réelle, un symbole, ...) ou complexe (un vecteur, un ensemble de symboles, ...),

- et ζ est une fonction

$$\zeta : e \in E \rightarrow \zeta(e)$$

qui associe à chaque arête un attribut.

Les fonctions ν et ζ peuvent être aussi indéfinies et dans ce cas on parle de graphe sans attributs. Dans le cas général, on parle de graphe avec attributs.

Étant donné une arête $e = (v_i, v_j)$, on dit que e connecte v_i avec v_j et v_i, v_j sont appelés *extrémités* de l'arête e .

La Fig. 2.1 montre un exemple de graphe sans attributs et la Fig. 2.2 montre un exemple de graphe avec attributs.

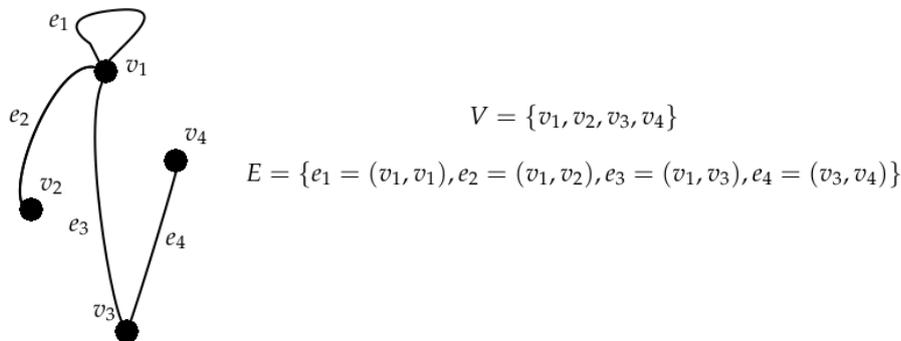


FIGURE 2.1 – Un graphe sans attributs

Rien n'exclut, comme on le voit dans la figure 2.1, que l'on puisse avoir des arêtes qui connectent un sommet à lui-même, ou d'avoir deux arêtes qui connectent les mêmes sommets. Néanmoins, dans le domaine de la reconnaissance de formes, on tend à utiliser des *graphes simples*, c'est-à-dire sans *boucles* (arêtes qui connectent un sommet à lui-même), ni *arêtes multiples* (arêtes qui connectent les mêmes sommets).

Les sommets extrêmes d'une arête e sont dits *incidentes* à l'arête e . Deux sommets incidents à la même arête sont dits *adjacents*. Egalement, deux arêtes incidentes à un même sommet sont dites *adjacentes*. Dans la Fig. 2.1 e_1 et e_2 sont adjacentes, ainsi que les sommets v_1 et v_3 ; v_3 et v_4 sont les extrémités de l'arête e_4 .

Le *degré* d'un sommet v ($d(v)$) et le nombre d'arêtes incidentes à v . Un *sommet pendant* est un sommet de degré 1; une arête incidente à un

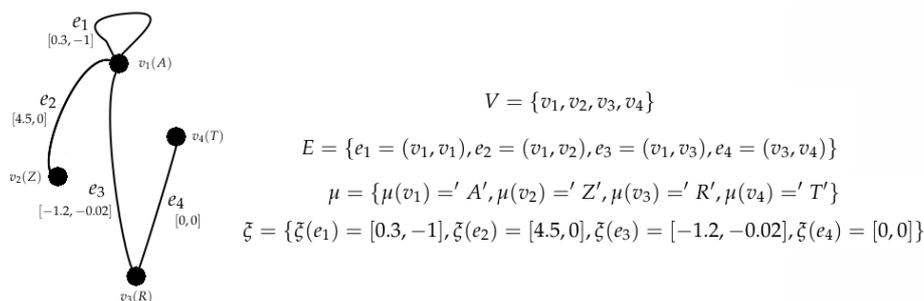


FIGURE 2.2 – Un graphe avec attributs

sommet pendant est dite pendante. Un sommet est dit *isolé* si son degré est 0. Le graphe de la Fig. 2.1 a deux sommets pendants (v_2 et v_4), un sommet de degré 2 et un sommet de degré 4 (par convention on compte deux fois une boucle).

Un graphe simple qui contient toutes les arêtes possibles entre les sommets est appelé un *graphe complet*.

2.2 GRAPHES ORIENTÉS ET NON ORIENTÉS

Dans certains problèmes, la relation entre les objets n'est pas symétrique. Dans ce cas on utilise des graphes orientés, dans lesquels les arêtes ont une orientation d'une de ces extrémités vers l'autre.

Définition 2.2.1 (Graphe orienté) *Un graphe orienté G est une quadruple $G(V, E, \nu, \xi)$, où V est un ensemble $\{v_1, v_2, \dots, v_n\}$ de sommets, E est un ensemble $\{e_k = (v_i, v_j) : v_i \text{ et } v_j \in V \text{ avec } k = 1, \dots, m\}$ d'arêtes, ν est une fonction*

$$\nu : v \in V \rightarrow \nu(v)$$

qui associe à chaque sommet un attribut qui peut être simple (une valeur réelle, un symbole, ...) ou complexe (un vecteur, un ensemble de symboles, ...), et ξ est une fonction

$$\xi : e \in E \rightarrow \xi(e)$$

qui associe à chaque arête un attribut.

Il faut ajouter que $(v_i, v_j) \neq (v_j, v_i)$ et que $(v_i, v_j) \in E \not\Rightarrow (v_j, v_i) \in E$

Donc, la définition de graphe orienté est très similaire à celle générale (qui définit un graphe non orienté), avec l'ajout d'une propriété qui contraint les arêtes à avoir une direction.

Dans un graphe orienté, on définit le *degré sortant* d'un sommet v , le nombre d'arêtes incidentes v qui partent de v (v est en première position dans la définition du couple de sommets extrêmes de l'arête) et *degré entrant* le nombre d'arêtes incidentes v qui arrivent dans v .

Définition 2.2.2 (Degrés d'un sommet) *Soit $G(V, E, \nu, \xi)$ un graphe orienté. Le degré entrant et le degré sortant d'un sommet sont définis comme suit.*

$$d^I(v) = |\{(v_i, v_j) \in E : v_j = v\}|$$

$$d^O(v) = |\{(v_i, v_j) \in E : v_i = v\}|$$

Dans la suite quand on parlera simplement de graphe, on considèrera les graphes non orientés, et à l'inverse, dans les cas spécifiques on parlera explicitement de graphes orientés. La Figure 2.3 montre un exemple de graphe orienté sans attributs.

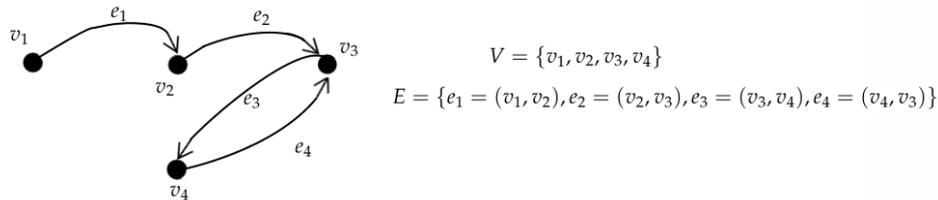


FIGURE 2.3 – Un graphe orienté sans attributs.

2.3 SOUS-GRAPHES

En Reconnaissance de Formes, parfois, on considère le graphe comme étant constitué de sous-parties connectées entre elles. Ainsi dans certains problèmes on cherche, par exemple, les sous-parties d'un objet dans une image, ou bien on essaye d'extraire des informations pertinentes à partir d'une combinaison de sous-parties d'un graphe, etc.

C'est pour cette raison qu'on introduit la définition de sous-graphe.

Définition 2.3.1 (Sous-graphe) *Un graphe $H(V_H, E_H, \mu_H, \xi_H)$ est un sous-graphe d'un graphe $G(V_G, E_G, \mu_G, \xi_G)$, notée $H \subseteq G$, si $V_H \subseteq V_G$ et $E_H \subseteq E_G$.*

On peut noter que dans cette définition on ne met pas en relation l'ensemble V_H et E_H , c'est-à-dire que si deux sommets de V_H sont les extrémités d'une arête e dans E_G , cela n'oblige pas à avoir $e \in E_H$. Cependant, dans certains problèmes on a la nécessité d'extraire d'un graphe un ensemble de sommets et toutes les arêtes qu'y sont incidentes. Dans ce cas on introduit une nouvelle définition.

Définition 2.3.2 (Sous-graphe induit) *Un sous-graphe $H(V_H, E_H, \mu_H, \xi_H)$ est dit sous-graphe induit d'un graphe $G(V_G, E_G, \mu_G, \xi_G)$, si H est un sous-graphe de G et $E_H = E_G \cap E(V_H)$.*

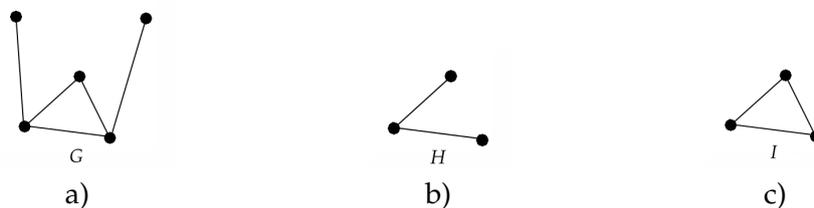


FIGURE 2.4 – Un graphe (a) et un exemple de sous-graphe (b) et sous-graphe induit (c).

La Figure 2.4 montre un exemple de graphe G sans attributs et d'un sous-graphe H et un sous-graphe induit I . De par la définition donnée, pour chaque sous-ensemble de sommets $V_H \subseteq V_G$ il existe et il est unique le sous-graphe induit correspondant.

Une autre définition nous intéressant est celle de sous-graphe couvrant, qui est, intuitivement un sous-graphe qui contient tous les sommets du graphe d'origine; par conséquence un sous-graphe couvrant ne contient pas toutes les arêtes du graphe original puisque autrement on aurait le même graphe initial. Voici la définition formelle.

Définition 2.3.3 (Sous-graphe couvrant) *Un sous-graphe $H(V_H, E_H, \mu_H, \xi_H)$ est dit sous-graphe couvrant d'un graphe $G(V_G, E_G, \mu_G, \xi_G)$, si H est un sous-graphe de G et $V_H = V_G$.*

Dans la Figure 2.5 on a un exemple de sous-graphe couvrant S du graphe G de la Figure 2.4a.

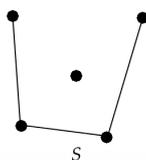


FIGURE 2.5 – Un sous-graphe couvrant correspondant au graphe G de la Figure 2.4a.

2.4 CHEMINS, CIRCUITS ET CONNEXITÉ

Définition 2.4.1 (Chemin) *Soit un graphe $G(V, E, \mu, \xi)$. Un chemin est une sequence finie alternée de sommets et d'arêtes :*

$$v_0 e_1 v_1 e_2, \dots, e_k, v_k$$

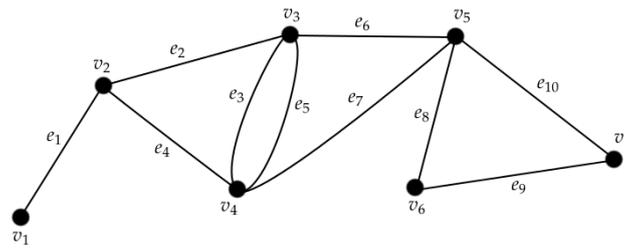
Un chemin commence toujours par un sommet, qui est nommé *sommet initial*, et se termine aussi par un sommet, qui est nommé *sommet terminal*. Le nombre k de sommets est dit la *longueur du chemin*. Un chemin est *ouvert* si $v_0 \neq v_k$, autrement il est *fermé*.

Un *chemin élémentaire* est un chemin dans lequel tous les sommets sont distincts. Un *chemin simple* est un chemin dans lequel toutes les arêtes sont distinctes. Dans un graphe simple, sans arêtes multiples, les chemins élémentaires coïncident avec les chemins simples. Un chemin élémentaire fermé est dit *circuit*.

Quelques exemple de chemins d'un graphe G sont fournis dans la Figure 2.6.

Une notion importante qui dérive de la notion de chemin est celle de *connexité*. Deux sommets u et v d'un graphe G sont connectés s'il existe un chemin entre u et v dans le graphe. Un graphe est *connexe* si tous les sommets sont connectés entre eux. Un sous-graphe connexe d'un graphe G est appelé une *composante connexe* du graphe.

Il faut noter que deux sommets connectés entre eux ne sont pas forcément adjacents; ils le sont si il existe un chemin de longueur 1 (une seule arête) entre les deux sommets; dans ce cas les deux sommets sont adjacents. Ainsi il ne faut pas confondre la notion de composante connexe avec celle de clique, qui sera utilisée dans le Chapitre 3. Une *clique* d'un graphe G est un sous-graphe dont les sommets sont tous adjacents entre eux. Une clique est aussi une composante connexe, mais l'inverse n'est pas



a) $v_1 e_1 v_2 e_2 v_3 e_3 v_4 e_4 v_2 e_2 v_3 e_6 v_5$

b) $v_2 e_4 v_4 e_7 v_5 e_{10} v_7$

c) $v_5 e_8 v_6 e_9 v_7 e_{10} v_5$

FIGURE 2.6 – Un graphe G et quelques exemple de chemins : un chemin (a), un chemin élémentaire (b) et un circuit (c).

vrai. Dans la Figure 2.7 nous montrons la différence entre une composante connexe et une clique.

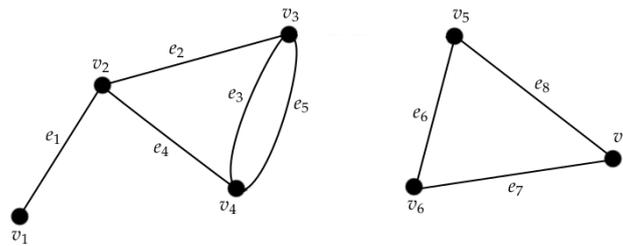


FIGURE 2.7 – Soit G le graphe de la Figure 2.6. Le sous-graphe induit par les sommets $\{v_1, v_2, v_3, v_4\}$ est une composante connexe du graphe mais elle n'est pas une clique, tandis que le sous-graphe induit par les sommets $\{v_5, v_6, v_7\}$ est une clique (et donc une composante connexe) de G .

APPARIEMENT DE GRAPHS

3

SOMMAIRE

| | | |
|-------|---|----|
| 3.1 | MÉTHODES D'APPARIEMENT | 15 |
| 3.1.1 | Définitions | 15 |
| 3.2 | ANALYSE DE L'ÉTAT DE L'ART | 17 |
| 3.2.1 | Algorithmes d'appariement basés sur les arbres de recherche | 17 |
| 3.2.2 | Autres techniques | 20 |
| 3.2.3 | Bilan de l'état de l'art | 21 |
| 3.3 | NOS CONTRIBUTIONS À L'APPARIEMENT DE GRAPHS | 22 |
| 3.3.1 | L'évaluation des algorithmes | 22 |
| 3.3.2 | L'appariement de pyramides de graphes | 26 |
| 3.4 | PERSPECTIVES | 27 |

CE chapitre introduit l'un des outils les plus importants utilisés en Reconnaissance de Formes Structurale : l'appariement de graphes. A partir de la fin des années soixante-dix, les techniques d'appariement de graphes ont été proposées comme un outil puissant de classification en RF. Dans ce scénario on a observé une augmentation considérable de propositions d'algorithmes de résolution. L'énorme quantité de matériel disponible et leur complexité a été souvent la principale cause de tentatives infructueuses d'utilisation de ces méthodes. Nous avons tenté de combler ce problème en faisant un compte-rendu détaillé de l'état de l'art des techniques de résolution des principaux problèmes de l'appariement de graphes, à savoir l'isomorphisme entre graphes, l'isomorphisme de sous-graphe, dans les versions exactes et inexactes, et la recherche du plus grand sous-graphe commun entre deux graphes (en anglais Maximum Common Subgraph - MCS). Cela a eu comme résultat la rédaction d'un article publié en 2004 [Conte et al. (2004)] dans la revue International Journal on Pattern Recognition and Artificial Intelligence (IJPRAI).

Après avoir illustré les principales méthodes d'appariement de graphes (Section 3.1), nous présentons donc, en Section 3.2, une synthèse de l'article cité en proposant, en plus, une mise à jour de son contenu sur la base des nouveaux articles sortis après sa parution.

Outre cette contribution, nous présentons deux autres contributions spécifiques en Section 3.3 : une procédure d'évaluation des algorithmes et une technique de matching entre pyramides de graphes.

Presque tous les problèmes abordés tombent dans la catégorie des problèmes NP-difficiles. Pour la résolution du MCS, nous avons aussi montré dans [(Conte et al. 2003; 2007)] qu'aujourd'hui il n'existe pas un algorithme de résolution plus efficace que les autres (Section 3.3.1), et nous avons donc montré l'importance d'avoir des données de Benchmark appropriées pour la comparaison et le choix des méthodes.

Une possible solution à la réduction des temps de calcul peut venir en utilisant les pyramides de graphes pour représenter les données du problème. Nous présentons dans la Section 3.3.2 un algorithme de matching entre pyramides de graphes qui permet de réduire les temps de calcul. Cet algorithme a été publié dans [Conte et al. (2005a; 2006)].

Enfin, la Section 3.4 décrit quelques perspectives sur le sujet de l'appariement de graphes.

3.1 MÉTHODES D'APPARIEMENT

Dans de nombreuses applications, une opération essentielle est la comparaison de deux objets ou d'un objet et d'un modèle. Lorsque l'information est représentée par des graphes, cette comparaison peut être effectuée en utilisant l'appariement de graphes. L'appariement de graphes est le processus de recherche d'une correspondance entre les sommets et les arêtes de deux graphes en s'assurant que les structures présentes dans un graphe soient préservées dans l'autre. L'appariement de graphes peut être dit "exact" quand on cherche une correspondance stricte entre les deux graphes ou entre leurs sous-parties, ou "inexact" quand une correspondance peut se produire même si les deux graphes comparés sont différents dans une certaine mesure (différence dans la structure ou différence entre les valeurs des attributs).

3.1.1 Définitions

Les algorithmes d'appariement considérés dans ce mémoire (car les plus utilisés en RF) sont : l'isomorphisme entre graphes, l'isomorphisme de sous-graphes, la recherche du plus grand sous-graphe commun à deux graphes et la distance d'édition entre graphes.

Isomorphisme

Définition 3.1.1 (Isomorphisme entre graphes) Soit $G(V, E, \mu, \xi)$ et $G'(V', E', \mu', \xi')$ deux graphes. Un isomorphisme entre G et G' est une fonction bijective $f : V \rightarrow V'$ telle que :

- $\mu(v) \equiv \mu'(f(v)) \forall v \in V$;
- pour chaque arête $e = (u, v) \in E$ elle existe une arête $e' = (f(u), f(v)) \in E'$ telle que $\xi(e) \equiv \xi'(e')$ et pour chaque $e' = (u', v') \in E'$ elle existe une arête $e = (f^{-1}(u'), f^{-1}(v')) \in E$ telle que $\xi(e) \equiv \xi'(e')$.

L'opérateur \equiv dépend des attributs attachés aux sommets et aux arêtes. Souvent pour des attributs symboliques l'équivalence se traduit en une égalité, mais l'équivalence peut être interprétée aussi comme appartenance à une même classe ; pour les attributs numériques l'équivalence est surtout basée sur une mesure de distance. Dans tous les cas, l'équivalence entre les attributs dépend fortement de l'application ; elle constitue une des questions fondamentales dans la représentation d'un problème par des graphes et elle influence beaucoup la pertinence des résultats.

Néanmoins, les algorithmes d'appariement, ne dépendent pas de la fonction d'équivalence des attributs si ce n'est pour le fait qu'une fonction d'équivalence coûteuse peut augmenter de beaucoup le temps d'exécution des algorithmes.

Isomorphisme de sous-graphe

Si $f : V \rightarrow V'$ est un isomorphisme entre les graphes G et G' , et G' est un sous-graphe induit de G , f est appelé isomorphisme de sous-graphe entre G et G' .

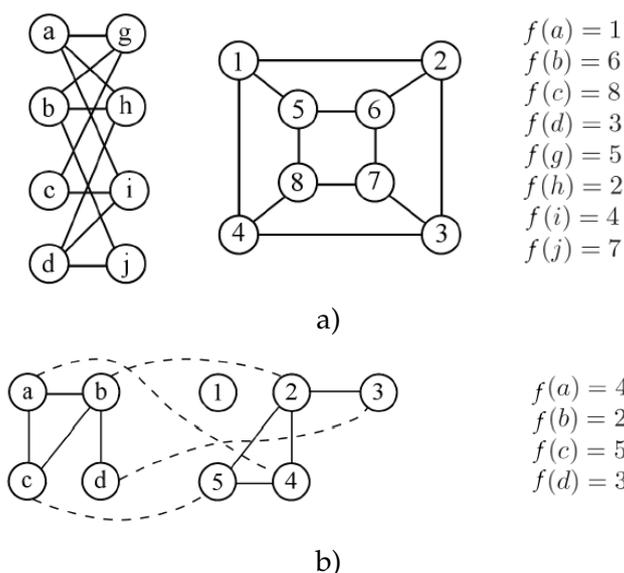


FIGURE 3.1 – Un isomorphisme entre deux graphes (a) et un isomorphisme de sous-graphe (b).

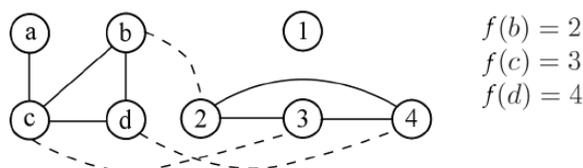


FIGURE 3.2 – Un exemple de plus grand sous-graphe commun à deux graphes.

Il a été prouvé que le problème de la recherche d'un isomorphisme de sous-graphe est un problème NP-difficile, tandis qu'il n'a pas été démontré la nature du problème de la recherche d'un isomorphisme entre graphes.

Dans la Figure 3.1 un exemple d'isomorphisme entre graphes et d'un isomorphisme de sous-graphes sont illustrés. Dans cet exemple nous n'avons pas inséré d'attributs sur les sommets et sur les arêtes ; par conséquent tous les sommets et toutes les arêtes des deux graphes sont équivalents entre eux.

Recherche du plus grand sous-graphe commun à deux graphes (MCS)

Définition 3.1.2 (Plus grand sous-graphe commun à deux graphes) Soit $G_1 = (V_1, E_1, \mu_1, \xi_1)$ et $G_2 = (V_2, E_2, \mu_2, \xi_2)$ deux graphes. Un sous-graphe commun de G_1 et G_2 , est un graphe $G = (V, E, \mu, \xi)$ tel que il existe un isomorphisme de sous-graphe entre G et G_1 , et entre G et G_2 . On appelle G le plus grand sous-graphe commun de G_1 et G_2 , $MCS(G_1, G_2)$, s'il n'existe aucun sous-graphe commun de G_1 de G_2 qui a un nombre de sommets plus grand que G .

Dans la Figure 3.2 on montre un exemple de MCS entre deux graphes.

Le problème de la recherche du plus grand sous-graphe commun à deux graphes peut être transformé en la recherche de la clique de dimension maximale sur le graphe d'association construit à partir des

deux graphes originaux. Le graphe d'association de deux graphes, est un graphe dont les sommets correspondent aux couples de sommets équivalents des deux graphes de départ ; ici la notion d'équivalence est la même de celle décrite pour l'isomorphisme de graphes. Les arêtes du graphe d'association (qui est un graphe non orienté) représentent la compatibilité entre les couples de sommets associés : un sommet du graphe d'association correspondant au couple (n_1, m_1) ($n_1 \in G_1$ et $m_1 \in G_2$), est connecté à un sommet correspondant au couple (n_2, m_2) , ssi il y a un isomorphisme entre le sous-graphe $\{n_1, n_2\}$ de G_1 et le sous-graphe $\{m_1, m_2\}$ de G_2 . Il peut être prouvé [Durand et al. (1999)] que chaque clique dans le graphe d'association correspond à un sous-graphe commun aux deux graphes d'origine et vice versa ; donc le plus grand sous-graphe commun à deux graphes peut être obtenu par la recherche de la clique de taille maximale dans le graphe d'association. La Figure 3.3 montre un exemple de clique du graphe d'association correspondant au plus grand sous-graphe commun aux deux graphes qui sont à l'origine du graphe d'association.

Distance d'édition entre graphes

L'idée de base de la distance d'édition entre graphes [Sanfeliu et Fu (1983), Neuhaus et al. (2006)] est la définition d'une dissimilarité entre deux graphes par le biais du nombre minimum de "distorsions" nécessaires pour transformer un graphe en un autre graphe. Le modèle de distorsion est défini par des opérations d'édition sur les sommets et sur les arêtes. Les opérations les plus utilisées sont l'insertion, la suppression et la substitution d'étiquettes sur les sommets et sur les arêtes. La séquence des opérations d'édition qui transforme un graphe dans un autre est appelée *chemin d'édition* (ou *edit path*). Chaque opération a un coût et la somme des coûts des opérations d'un chemin d'édition représente le coût du chemin. Finalement, la *distance d'édition* entre deux graphes est défini comme le coût du plus petit chemin d'édition entre les deux graphes.

La Figure 3.4 présente un exemple de chemin d'édition entre deux graphes.

La résolution du problème de la distance d'édition entre graphes est NP-difficile.

3.2 ANALYSE DE L'ÉTAT DE L'ART

Nombreux sont les articles qui traitent de l'appariement de graphes parus dans les dernières décennies (voir Table 3.1). Dans cette section, on présente une synthèse critique de l'état de l'art du domaine.

3.2.1 Algorithmes d'appariement basés sur les arbres de recherche

Beaucoup d'algorithmes pour la résolution des problèmes d'appariement sont basés sur l'exploration d'arbres de recherche avec *backtracking*. L'idée de base est qu'un appariement partiel (initialement vide) est étendu, de manière itérative, par insertion d'un nouveau couple de sommets ; le couple est choisi sur la base de certaines conditions qui assurent la compatibilité du nouveau couple avec l'ensemble de couples déjà présents

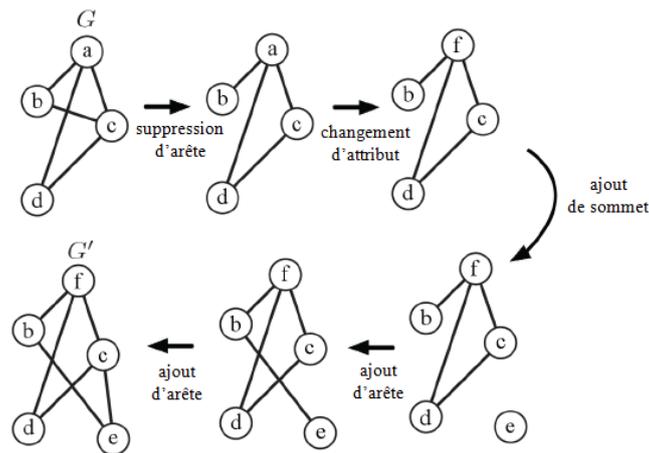


FIGURE 3.4 – Un exemple de chemin d'édition entre deux graphes, les lettres dans les sommets représentent les attributs.

| Période | Nombres d'articles |
|--------------|--------------------|
| jusqu'à 1990 | 4 |
| 1991-1994 | 7 |
| 1995-1998 | 29 |
| 1999-2002 | 51 |
| 2003-2005 | 12 |
| 2006-2008 | 9 |
| 2009-2012 | 24 |
| 2013-2014 | 10 |

TABLE 3.1 – Nombre d'articles publié dans des conférences et revues internationales, qui présentent une technique d'appariement de graphes.

dans l'appariement ; de plus le meilleur couple est choisi en fonction de certaines heuristiques qui facilitent la coupure le plus tôt possible des branches de l'arbre qui ne pourraient pas donner des bonnes solutions.

Une fois un nouveau couple ajouté, l'algorithme trouve un appariement complet ou atteint un point où l'appariement partiel courant ne peut plus être étendu à cause des contraintes d'appariement. Dans ce dernier cas, l'algorithme effectue un *backtracking*, c'est-à-dire annule la dernière insertion jusqu'à ce qu'il trouve une correspondance partielle pour lesquelles une extension alternative de l'appariement est possible.

Si toutes les correspondances possibles qui satisfont les contraintes ont déjà été essayées, l'algorithme s'arrête.

Plusieurs stratégies de mise en œuvre de cet algorithme ont été utilisées, se différenciant entre elles par l'ordre avec lequel les correspondances partielles sont visitées. Parmi les plus utilisées il y a la recherche en profondeur (*Depth First Search*), la recherche en largeur (*Breadth First Search*) ou la recherche par le "meilleur-d'abord" (*Best First Search*). La recherche *Best First Search* avec une bonne définition de l'heuristique associée à cette stratégie de recherche, donne souvent les meilleurs résultats. L'algorithme peut-être le plus connu dans la catégorie de la recherche *Best First Search* est l'algorithme A^* [Hart et al. (1968)].

Une propriété intéressante des techniques basées sur les arbres de re-

cherche est qu'elles peuvent être très facilement adaptées afin de prendre en compte les attributs sur les sommets et sur les arêtes sans aucune limitation sur le type d'attributs qui peuvent être utilisés. Cette propriété est très importante dans la RF où les attributs jouent un rôle clé dans l'efficacité d'une représentation.

Parmi les algorithmes les plus connus, puisqu'ils ont montré les meilleures performances par rapport aux autres algorithmes, on cite [Cordella et al. (2001), Solnon (2010)] pour la solution du problème de l'isomorphisme entre graphes et de l'isomorphisme de sous-graphe. Cordella et al. (2001) utilise certaines propriétés sur le voisinage des sommets pour réduire l'espace de recherche pour la résolution de l'isomorphisme entre graphes. Solnon (2010) modélise le problème de l'isomorphisme de sous-graphe à travers la formulation d'un problème de satisfaction de contraintes et introduit une contrainte qui mène à un algorithme bien plus efficace que les autres.

Pour la résolution du problème de la recherche du plus grand sous-graphe commun, les algorithmes les plus représentatifs des techniques basées sur l'exploration de l'arbre de recherche sont publiés dans [McGregor. (1982), Durand et al. (1999), Balas et Yu (1986)]. Les trois stratégies se différencient dans la modalité de coupe de l'arbre de recherche, les deux premières propositions basent leur stratégie sur la taille du sous-graphe maximal de la solution courante et la troisième reformule le problème par un problème de coloriage des sommets (c'est à dire la recherche de la manière de colorier les différents sommets d'un graphe de sorte qu'aucun couple de sommets adjacents ait ses extrémités de la même couleur).

Les méthodes basées sur les arbres de recherche ont été utilisées aussi pour l'appariement inexact. On peut citer, par exemple, [Cook et al. (2003)] qui propose un algorithme basé sur les arbres de recherche pour la distance d'édition entre graphes, ou l'algorithme de Tsai et Fu (1983) pour la résolution de l'isomorphisme inexact de sous-graphe.

3.2.2 Autres techniques

A cause de la complexité élevée des algorithmes basés sur les arbres de recherche, et pour que les méthodes d'appariement puissent être utilisées dans la pratique dans les applications de RF, d'autres techniques ont été développées.

Ces techniques sont souvent des solutions sous-optimales aux problèmes d'appariement. Soulignons, qu'il ne faut pas confondre la sous-optimalité avec l'inexactitude ; l'inexactitude signifie que dans la définition même de l'appariement, on ne cherche pas la correspondance exacte entre deux graphes, mais que l'on cherche une solution qui minimise cette différence (optimalité). Par contre les algorithmes sous-optimaux peuvent être aussi appliqués aux problèmes d'appariement exacts, mais souvent ils trouvent une solution qui n'est pas la meilleure du point de vue de l'appariement. Donc, on peut dire que, si avec les algorithmes inexacts une correspondance non parfaite est recherchée, dans les algorithmes sous-optimaux la solution "inexacte" est un "moindre mal", qu'on accepte pour avoir un temps de réponse raisonnable.

Les méthodes d'appariement examinées jusqu'à présent exploitent une formulation des problèmes d'appariement directement en termes de graphes. Une approche radicalement différente consiste à transformer le problème d'appariement de graphes, qui est en soi un problème d'optimisation discrète, en un problème d'optimisation continue. Ces algorithmes ne garantissent pas l'optimalité de la solution (bien que les plus sophistiqués d'entre eux comprennent des techniques pour éviter des optima locaux triviaux). En outre, la solution trouvée doit être reconvertie du domaine continu au domaine initial par un processus qui peut introduire un niveau supplémentaire d'approximation. Néanmoins, dans de nombreux contextes d'applications cette approche est très attrayante en raison de son coût de calcul très réduit qui est généralement polynomial (et avec un faible exposant) avec la taille des graphes. En plus, dans de nombreux cas, la solution est construite par des améliorations successives à partir d'un appariement initial, permettant à l'utilisateur de choisir entre une solution rapide et inexacte et une plus coûteuse mais plus précise. Dans cette catégorie, nous citons les méthodes très connues de Christmas et al. et de Myers et al. [R. Myers et Hancock (2000), W. J. Christmas et Petrou (1995)] et les travaux plus récents basés sur la programmation linéaire pour la résolution de l'appariement de sous-graphes et la distance d'édition entre graphes [Justice et Hero (2006), LeBodic et al. (2012)].

Une autre technique pour la résolution de l'appariement entre graphes est appelée technique spectrale puisqu'elle est basée sur l'idée suivante : les valeurs propres et les vecteurs propres de la matrice d'adjacence d'un graphe sont invariants par rapport à des permutations de sommets. Par conséquent, si deux graphes sont isomorphes, leurs matrices d'adjacence auront les mêmes valeurs et vecteurs propres. Malheureusement, l'inverse n'est pas vrai : nous ne pouvons pas déduire de l'égalité des valeurs propres/vecteurs propres que deux graphes sont isomorphes. Cependant, comme le calcul de valeurs propres/vecteurs propres est un problème bien étudié, qui peut être résolu en temps polynomial, il y a un grand intérêt dans l'utilisation de cette méthode pour l'appariement de graphes. Une limitation importante de ces méthodes est qu'elles sont d'ordre purement structurel, dans le sens où ils ne sont pas en mesure d'exploiter les attributs des sommets et des arêtes. C'est une limitation forte puisque nous avons vu que souvent dans les applications de RF, les attributs contiennent des informations très pertinentes pour le processus d'appariement. En outre, certaines des méthodes spectrales sont en mesure de traiter uniquement des attributs réels affectés qu'aux arêtes en utilisant une matrice d'adjacence avec des éléments à valeurs réelles. Les travaux les plus représentatifs de cette catégorie sont les travaux de Hancock et ses collègues [Robles-Kelly et Hancock (2005), Qiu et Hancock (2006)] et les travaux de Caelli et al. [Caelli et Kosinov (2004)].

3.2.3 Bilan de l'état de l'art

Les techniques de résolution pour les différents problèmes d'appariement sont nombreuses et variées et il est donc difficile de les catégoriser et d'évaluer de manière pertinente. Le défi commun à toutes ces proposi-

tions est représenté par la réduction des temps de calculs. Des nouvelles méthodes apparaissent souvent dans des revues et conférences internationales. Par contre chaque méthode utilise des bases de données différentes et la comparaison avec l'état de l'art n'est pas souvent présente dans ces articles.

Certaines tentatives ont été faites dans la direction de proposer des benchmark pour la comparaison d'algorithmes pour l'isomorphisme de graphes [Foggia et al. (2001b)], mais elles restent à compléter.

Dans la prochaine section, nous allons illustrer notre contribution à l'évaluation des algorithmes de MCS et proposer une approche originale pour faire face au problème des temps de calculs dans l'appariement de graphes.

3.3 NOS CONTRIBUTIONS À L'APPARIEMENT DE GRAPHES

3.3.1 L'évaluation des algorithmes

Les problèmes d'appariement, comme on l'a déjà dit, sont pour la plupart des problèmes NP-difficiles. Il est important donc d'évaluer, parmi tous les algorithmes présents dans la littérature, ceux qui sont les plus efficaces.

Les auteurs de chaque nouvel algorithme fournissent généralement des résultats expérimentaux qui démontrent, sous des hypothèses convenables, que leur méthode améliore les précédentes. Néanmoins, il est toujours très difficile pour un utilisateur de choisir l'algorithme qui est le mieux adapté pour traiter son problème. En fait, des résultats de performance d'un algorithme sur certains cas de test spécifiques peuvent souvent ne fournir aucun indice utile sur ce qui se passera dans un domaine différent.

Ainsi, il est important de pouvoir évaluer les performances de manière objective et ce problème a donc constitué un objectif de mes travaux de recherche.

Le premier type de comparaison qui peut être facilement effectué entre les différents algorithmes est une comparaison théorique. En fait, il est toujours possible d'estimer la complexité de calcul de chaque algorithme dans le pire des cas et dans le meilleur des cas. Un problème commun est que les utilisateurs d'algorithmes d'appariement de graphes doivent choisir un algorithme qui correspond le mieux à leur problème. Dans de nombreux cas, cependant, une description d'un algorithme à travers son comportement dans le meilleur ou le pire des cas reste insuffisante. On peut supposer, par exemple, que les algorithmes A_1 et A_2 sont disponibles pour résoudre un problème d'appariement de graphe donné, et que l'algorithme A_1 est plus rapide que l'algorithme A_2 dans le meilleur des cas et dans le pire des cas. Est-ce suffisant pour préférer l'algorithme A_1 à l'autre ? Bien sûr, il ne l'est pas. L'utilisateur a besoin de plus de détails sur le comportement des deux algorithmes, pour choisir le meilleur. En particulier, l'information dont un utilisateur pourrait avoir besoin est : quel algorithme est plus rapide sur les graphes décrivant mon problème ? La réponse à cette question n'est pas simple du tout.

Etant donné que l'information concernant la complexité dans le

| ORGANISATION DE LA BASE DE DONNEES (81 400 couples de graphes) | Categories de graphes | Taille du MCS | | | | |
|---|-----------------------|---------------|------|------|------|------|
| | | | 10% | 30% | 50% | 70% |
| Graphes aléatoires (18 600 couples) | $\eta=0.01$ | 100 | 100 | 100 | 100 | 100 |
| | $\eta=0.05$ | 1100 | 1100 | 1100 | 1100 | 1100 |
| | $\eta=0.1$ | 1100 | 1300 | 1300 | 1300 | 1300 |
| | $\eta=0.2$ | 1100 | 1300 | 1300 | 1300 | 1300 |
| Maillages Réguliers (8 100 couples) | 2D | 700 | 700 | 1000 | 1000 | 1000 |
| | 3D | 700 | 700 | 400 | 400 | 400 |
| | 4D | 100 | 500 | 300 | 100 | 300 |
| Maillages Irréguliers (23 900 couples) | 2D $r=0.2$ | 400 | 700 | 900 | 1000 | 1000 |
| | 2D $r=0.4$ | 700 | 700 | 1000 | 1000 | 1000 |
| | 2D $r=0.6$ | 700 | 700 | 1000 | 1000 | 1000 |
| | 3D $r=0.2$ | 500 | 700 | 400 | 400 | 400 |
| | 3D $r=0.4$ | 500 | 700 | 400 | 400 | 400 |
| | 3D $r=0.6$ | 500 | 700 | 400 | 400 | 400 |
| | 4D $r=0.2$ | 100 | 500 | 300 | 100 | 300 |
| | 4D $r=0.6$ | 100 | 500 | 300 | 100 | 300 |
| Graphes à Valence Limité (15 400 couples) | $n=3$ | 700 | 1200 | 1300 | 1300 | 1300 |
| | $n=6$ | 400 | 1000 | 1200 | 1300 | 1300 |
| | $n=9$ | 100 | 800 | 1100 | 1200 | 1200 |
| Graphes à Valence Limité (Irrégulier) (15 400 couples) | $n=3$ | 700 | 1200 | 1300 | 1300 | 1300 |
| | $n=6$ | 400 | 1000 | 1200 | 1300 | 1300 |
| | $n=9$ | 100 | 800 | 1100 | 1200 | 1200 |

FIGURE 3.5 – L'organisation de la base de données de graphes crée pour l'évaluation des algorithmes de MCS : pour chaque typologie de graphe, pour chaque taille de sous-graphe commun et pour chaque valeur des paramètres, il est indiqué le nombre de graphes générés [Conte et al. (2007)].

meilleur et dans le pire des cas n'est pas suffisante pour la comparaison d'algorithmes, un autre paramètre qui peut être utilisé est la complexité de calcul dans le cas moyen. En effet, même si la complexité dans le pire des cas de l'algorithme A_2 est supérieure à la complexité de A_1 , A_2 pourrait être plus rapide de A_1 dans le cas moyen. Malheureusement, la complexité dans le cas moyen peut être déterminée analytiquement pour des algorithmes simples, mais cela peut se révéler une tâche difficile pour de nombreux algorithmes d'appariement de graphes.

La seule possibilité restante est alors d'effectuer une large comparaison expérimentale de différents algorithmes, pour mesurer leurs performances sur une grande base de données contenant de nombreuses catégories de graphes. Malheureusement, seuls quelques articles se sont confrontés au problème d'une comparaison approfondie des algorithmes d'appariement de graphes [Bunke et Vento (1999), Cordella et al. (1999)]. Ainsi, il semble que l'habitude de proposer de nouveaux algorithmes prévaut sur la nécessité d'évaluer le rendement de ceux qui existent déjà de manière objective. En conséquence, les utilisateurs des approches basées sur graphes ne peuvent utiliser que des critères qualitatifs pour sélectionner l'algorithme qui semble mieux répondre aux contraintes de l'application. Il n'y a que peu ou pas d'information sur le comportement de ces algorithmes en fonction du type et de la taille des graphes.

Afin de combler ce manque nous avons réalisé (et publié dans Conte et al. (2007)) une activité de caractérisation des performances d'un ensemble d'algorithmes pour la recherche du plus grand sous-graphe commun à deux graphes. Dans ce but, nous avons aussi construit une grande base de données qui est caractéristique des typologies de graphes utilisés dans les applications (décrites ci-après). Ainsi notre contribution au sujet de l'évaluation des algorithmes de MCS est double :

- la création d'une base de données de graphes caractérisant les typologies utilisés dans les applications a été faite. Dans d'autres domaines de recherche (par exemple l'OCR), la disponibilité de grandes bases de données standard améliore la vérifiabilité et la comparabilité des résultats expérimentaux de chaque méthode. La création d'une base de données de graphes n'est certainement pas une tâche simple. Une des questions les plus importante à résoudre est celle de décider si les graphes doivent être collectés à partir des applications du monde réel ou s'ils doivent être générés de manière synthétique, selon certains modèles probabilistes. Ce dernier choix, en plus d'être simple à mettre en œuvre, permet un contrôle plus précis sur les caractéristiques des graphes ; les modèles pour la génération de graphes synthétiques ont été dérivés de l'analyse des graphes dans des applications réelles ;
- l'évaluation détaillée de trois algorithmes de résolution du MCS sur la base de données créée a aussi été réalisée.

Nous avons créé une grande base de données contenant différentes catégories de graphes avec attributs, à savoir graphes aléatoires, maillages réguliers et irréguliers, 2D, 3D et 4D, graphes à valence bornée. Pour chaque catégorie, nous avons généré des paires de graphes ayant un sous-graphe commun maximal connu.

Les *graphes aléatoires* sont des graphes dans lesquels les arêtes relient

| | | <table border="1" style="margin: auto;"> <tr><td>Mc Gregor</td></tr> <tr><td>Durand Pasari</td></tr> <tr><td>Basel Von</td></tr> </table> | | | | | | | Mc Gregor | Durand Pasari | Basel Von |
|---------------|-----|---|----|----|----|----|----|----|-----------|---------------|-----------|
| Mc Gregor | | | | | | | | | | | |
| Durand Pasari | | | | | | | | | | | |
| Basel Von | | | | | | | | | | | |
| | | Regular Bounded Graphs | | | | | | | | | |
| | | N | | | | | | | | | |
| | | 10 | 15 | 20 | 25 | 30 | 35 | 40 | | | |
| v | A | | | | | | | | | | |
| | 3 | 33% | 0 | 1 | 1 | 1 | 2 | 3 | 4 | | |
| 50% | | 0 | 0 | 1 | 2 | 3 | 3 | 4 | | | |
| 75% | | 0 | 0 | 1 | 2 | 2 | 3 | 3 | | | |
| 6 | 33% | 1 | 1 | 1 | 1 | 1 | 0 | 3 | | | |
| | 50% | 0 | 0 | 0 | 0 | 0 | 3 | 3 | | | |
| | 75% | 0 | 0 | 1 | 1 | 1 | 1 | 2 | | | |
| 9 | 33% | 0 | 1 | 1 | 1 | 1 | 2 | 3 | | | |
| | 50% | 0 | 0 | 0 | 0 | 0 | 3 | 1 | | | |
| | 75% | 0 | 0 | 1 | 1 | 1 | 0 | 2 | | | |

FIGURE 3.6 – Un exemple de comparaison des algorithmes de MCS considérés pour une typologie de graphe donnée (graphe irréguliers à valence limitée); chaque élément du tableau indique quel algorithme est le plus rapide et de quel ordre de grandeur il est supérieur aux autres [Conte et al. (2007)].

des sommets sans une régularité définie. Ils ont été introduits pour simuler les applications dans lesquelles les entités (représentées par des sommets) peuvent établir des relations (représentées par les arêtes) avec n'importe quelle autre entité. Cette hypothèse se produit généralement dans les niveaux de traitement intermédiaires et supérieurs en vision par ordinateur.

Les *maillages réguliers 2D, 3D, et 4D* ont été introduits pour simuler des applications portant sur les structures régulières comme celles fonctionnant aux niveaux inférieurs du traitement en vision par ordinateur. Les *maillages irréguliers* simulent les représentations de structures régulières légèrement déformées à cause du bruit.

Les *graphes à valence bornée* modèlent les applications dans lesquelles chaque objet a un nombre fixe de relations avec d'autres objets. On retrouve ce type de graphes surtout dans les applications de chimio-informatique. Pour cette typologie de graphes, on a introduit aussi une version irrégulière c'est-à-dire des graphes dans lesquels certains sommets n'ont pas exactement le nombre fixé d'arêtes.

La Figure 3.5 fourni un résumé des caractéristiques des graphes de la base de données générée.

Avec cette base de données nous avons pu évaluer certains des algorithmes les plus représentatifs du problème de la recherche du plus grand sous-graphe commun à deux graphes et nous avons montré (comme nous le pensons de manière intuitive) que, au moins pour le problème considéré, aucun algorithme peut être considéré comme le meilleur par rapport aux autres, mais que le choix de l'algorithme dépend du type des graphes qui font l'objet de l'appariement. Dans la Figure 3.6, on présente un exemple de comparaison des algorithmes considérés, pour une catégorie donnée de graphes. On peut remarquer la variabilité des résultats en fonction des différentes valeurs des paramètres structurels des graphes utilisés. L'expérimentation complète sur toute la base de données est présentée dans Conte et al. (2007); cet article peut être très utile pour un

utilisateur qui connaît bien la structure des graphes de son problème et veut choisir le meilleur algorithme d'appariement adapté à son cas.

Sur ce travail nous pouvons faire la considération suivante : dans les graphes créés pour la base de données, les attributs ne sont pas suffisamment présents. Cependant la présence et la nature des attributs influencent beaucoup les performances des algorithmes. Pour les attributs il faudrait ajouter des catégories (attributs numériques, symboliques, etc.) et élargir la base de données avec l'ajout de graphes attribués.

3.3.2 L'appariement de pyramides de graphes

Le problème de la définition d'un algorithme de recherche d'appariement de graphes est encore ouvert et l'état de l'art nous montre que des nouveaux algorithmes apparaissent régulièrement essayant d'améliorer les performances des précédents.

Afin de lever le verrou principal des algorithmes actuels (temps de calcul), nous avons abordé le problème d'un point de vue différent en cherchant à exploiter les propriétés de structures de données particulières pour accélérer le processus d'appariement. C'est le cas, par exemple, des graphes planaires pour lesquels un algorithme polynomial pour la résolution de l'isomorphisme de sous-graphes a été trouvé [de la Higuera et al. (2013)].

De manière similaire nous avons présenté un algorithme d'appariement de pyramides de graphes qui a pour but d'accélérer la comparaison entre deux pyramides. Les pyramides de graphes ont été développées de manière approfondie dans les années 1990 surtout dans le cadre de la segmentation et de l'analyse d'images (voir par exemple Jolion et Montanvert (1992), Kropatsch (1997), Meer (1989)), mais un appariement entre pyramides n'avait jamais été proposé jusqu'en 2006 quand nous avons publié notre proposition ((Conte et al. 2006)).

L'algorithme proposé résout une variante des problèmes d'appariement définis auparavant et il a été conçu dans le cadre d'une application d'analyse vidéo. Néanmoins, l'idée à la base de l'algorithme reste valide et applicable à d'autres contextes applicatifs et à d'autres problèmes d'appariement.

Commençons par définir qu'est-ce qu'une pyramide irrégulière de graphes [Jolion et Montanvert (1992)] :

- la pyramide est un ensemble de niveaux L_1, L_2, \dots, L_k ;
- chaque niveau L_i est représenté par un graphe non orienté $G_i(N_i, E_i)$;
- pour chaque niveau L_i tel que $i > 1$ il est définie une fonction surjective p qui associe chaque sommet $n \in N_i$ avec un sommet dans N_{i-1} , appelé parent de n :

$$p : n \in N_i \rightarrow p(n) \in N_{i-1}$$

$$\forall n \in N_{i-1}, \exists m \in N_i : p(m) = n$$

- pour chaque niveau L_i tel que $i > 1$, si deux sommets sont adjacents, ou ils ont le même parent ou bien ses parents sont adjacents ; si deux

sommets ne sont pas adjacents, ils ne peuvent pas avoir le même parent :

$$(n, m) \in E_i \Rightarrow p(n) = p(m) \vee (p(n), p(m)) \in E_{i-1}$$

$$(n, m) \notin E_i \Rightarrow p(n) \neq p(m)$$

Pour chaque niveau L_i tel que $i < k$ on peut aussi définir une fonction *child* qui associe chaque sommet $n \in N_i$ avec ces enfants, i.e. les sommets dans N_{i+1} qui ont n comme parent :

$$\text{child} : n \in N_i \rightarrow \{m \in N_{i+1} : p(m) = n\}$$

Intuitivement, les enfants d'un sommet représentent une décomposition en parties de l'information associée à leur parent.

Le principe de la méthode proposée est le suivant : avec une représentation hiérarchique, on peut exploiter les relations parents/enfants pour accélérer le processus d'appariement. En effet la logique de l'algorithme est d'essayer d'apparier les graphes des niveaux les plus hauts de la pyramide dans lesquels les graphes sont petits et l'appariement est rapide à déterminer. Si le problème est résolu à ces niveaux, il n'y a pas besoin d'aller aux niveaux inférieurs puisque on peut propager les correspondances trouvées à un niveau aux niveaux inférieurs à travers les liens de parenté. S'il y a des sommets pour lesquels un appariement n'a pas été trouvé, on considère le niveau suivant de la pyramide uniquement pour les sous-graphes qui n'ont pas été apparés. En conséquence, même si dans les niveaux inférieurs les graphes ont une taille plus importante, le fait de considérer que certains sous-graphes réduit la taille du problème et l'appariement peut encore être rapide. Une vue d'ensemble de la méthode est présentée dans la Figure 3.7.

L'algorithme actuel se concentre surtout sur l'appariement des sommets et seules les relations hiérarchiques entre sommets sont exploitées. Une version améliorée de l'algorithme devrait introduire des contraintes sur les arêtes des graphes dans les différents niveaux de la pyramide, pour réduire encore plus les temps de calcul.

3.4 PERSPECTIVES

Dans ce chapitre, nous avons illustré les problèmes de la recherche de l'isomorphisme entre graphes, l'isomorphisme de sous-graphe, la recherche du plus grand sous-graphe commun à deux graphes et la distance d'édition entre graphes. Ces problèmes sont à la base de nombreuses applications de Reconnaissance de Formes.

Notre contribution à la thématique de l'appariement de graphes a été tout d'abord la proposition d'un cadre de référence pour la comparaison/évaluation des techniques de résolution de l'appariement. Notre proposition a contribué à bien catégoriser les nombreuses propositions parues dans les années passées et à aider les chercheurs à avoir une meilleure idée de l'intérêt des méthodes existantes pour en proposer de nouvelles

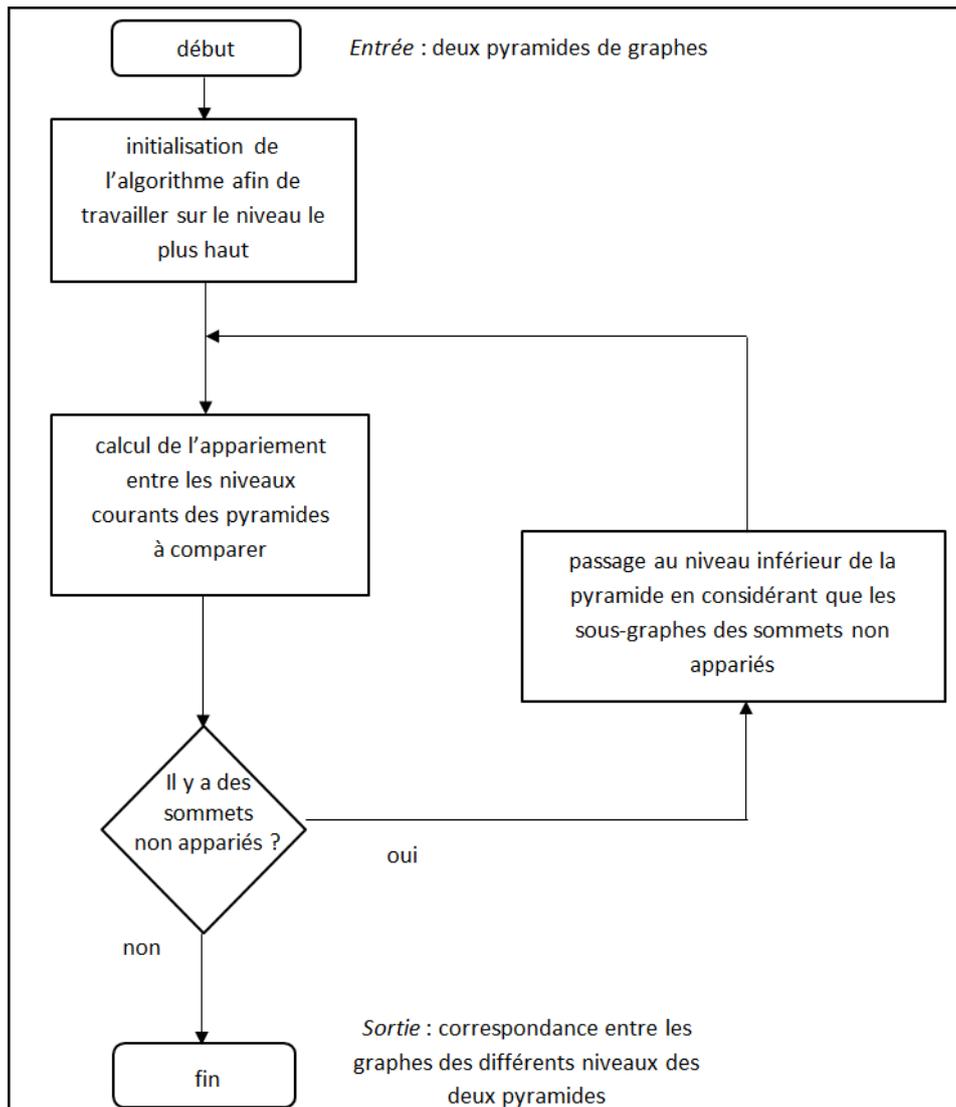


FIGURE 3.7 – Vue d'ensemble de la méthode d'appariement basée sur une représentation pyramidale des données.

à partir des résultats déjà acquis. Une perspective, dans cette direction, pourrait être de recenser tous les algorithmes pour la distance d'édition entre graphes puisque est l'une des plus étudiées dans la communauté de la RF structurelle.

Une deuxième contribution a été l'évaluation des principaux algorithmes de MCS : l'importance de ce travail réside dans le fait d'avoir défini un benchmark de référence pour évaluer les futures propositions ; la base de données construite étant représentative des typologies de graphes utilisés dans les applications, cela rend facile la tâche du choix de l'algorithme d'appariement le plus approprié pour une application donnée. Pour cela, il y a plusieurs perspectives envisageables : la première est d'étendre la base de données à d'autres catégories de graphes qui progressivement se présentent dans les applications. On peut envisager aussi d'ajouter d'autres algorithmes pour avoir une comparaison plus large. La deuxième perspective intéressante serait d'élargir ce travail d'évaluation aux méthodes basées sur la distance d'édition de graphes. En effet, comme nous l'avons dit, dans les dernières années, on constate une explosion du nombre de propositions de résolution du problème de la distance d'édition, sans pourtant une évaluation systématique de ces méthodes.

Enfin, nous avons proposé un algorithme d'appariement entre pyramides de graphes qui, en exploitant une représentation hiérarchique rend plus rapide l'appariement de graphes correspondants à des contenus d'images. Cette méthode a été définie dans un cadre applicatif spécifique, et une perspective pourrait être d'examiner son comportement dans d'autres domaines. De même, une perspective envisageable est de proposer une distance d'édition entre pyramides : les relations hiérarchiques pourraient être exploitées pour propager les coûts d'édition aux différents niveaux de la pyramide.

PLONGEMENT DE GRAPHES

4

SOMMAIRE

| | | |
|-------|---|----|
| 4.1 | LE PLONGEMENT EXPLICITE : LE GRAPH EMBEDDING | 33 |
| 4.2 | LE PLONGEMENT IMPLICITE : LES NOYAUX DE GRAPHES | 34 |
| 4.3 | CONTRIBUTIONS AU PLONGEMENT DE GRAPHES | 35 |
| 4.3.1 | Des noyaux de graphes | 35 |
| 4.3.2 | Comparaison entre plongement implicite et explicite . . . | 37 |
| 4.4 | PERSPECTIVES | 40 |

PARMI les techniques d'utilisation des graphes dans la RF, le plongement de graphes est peut-être la nouveauté la plus intéressante et la plus importante, parce que ces techniques permettent à la fois d'exploiter la richesse de l'information contenue dans la représentation structurale et la robustesse des méthodes bien connues de classification et apprentissage statistique.

Ce chapitre illustre les techniques de plongement explicite de graphes (ou *Graph Embedding*) et de plongement implicite (noyaux de graphes). Dans la Section 4.1, nous définissons le Graph Embedding et présentons un état de l'art sur le sujet, tandis que dans la Section 4.2 une synthèse de l'état de l'art sur les noyaux de graphes est proposée.

J'ai abordé, au travers de nombreuses collaborations de recherche le sujet du plongement de graphes : en collaboration avec Luc Brun, Professeur à l'ENSICAEN, j'ai pu étudier certains noyaux de graphes : nous avons proposé plusieurs noyaux de graphes dans différentes applications. Durant cette collaboration, j'ai co-encadré différents étudiants de l'ENSICAEN en stage dans mon laboratoire en Italie. Le résultat de cette collaboration a été la publication d'articles qui présentent des noyaux de graphes appliqués à la biochimie (Brun et al. (2010)) ou à l'analyse de vidéos (Brun et al. (2011a;b), Mahboubi et al. (2013b;a; 2014)). Une discussion sur ces contributions aux noyaux de graphes est faite dans la Section 4.3.1.

Il n'est pas aisé de déterminer, entre les deux techniques de plongement, laquelle peut fournir les meilleurs résultats. Afin de comparer les deux techniques, nous avons mis en place une collaboration entre différents groupes de recherche qui travaillent sur le plongement de graphes. Ce travail, qui a été publié dans le Workshop GbR 2013 (Conte et al.

(2013b)), a été réalisé avec la collaboration de Muhammad Muzzamil Luqman du laboratoire L3i de La Rochelle, Jean-Yves Ramel et Nicolas Sidère du laboratoire LI de Tours, Luc Brun, Benoit Gauzère et Jaume Gibert du laboratoire GreyC de Caen et avec Mario Vento du laboratoire MIVIA de Salerno (Italie). Ce travail est décrit dans la Section 4.3.2.

La dernière section (Section 4.4) est dédiée aux perspectives que j'envisage d'explorer dans la thématique du plongement de graphes.

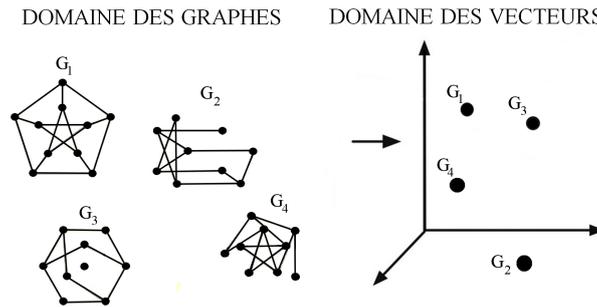


FIGURE 4.1 – Illustration du plongement explicite.

4.1 LE PLONGEMENT EXPLICITE : LE GRAPH EMBEDDING

Le plongement de graphes vise à transformer les données structurelles dans un espace vectoriel doté de métriques qui permet d'utiliser, pour des applications de classification et d'apprentissage, les outils de reconnaissance de formes statistiques bien connus et robustes.

L'idée principale à la base de cette technique est que la transformation des graphes en vecteurs doit permettre d'exploiter en même temps la richesse d'information portée par les graphes (la transformation étant conçue de manière à ce que les vecteurs préservent le mieux possible les contraintes structurelles intrinsèques aux graphes) et la robustesse des techniques de classification statistique.

Ainsi, le Graph Embedding est une technique qui fait correspondre des graphes à des points dans un espace vectoriel, de manière à ce que des graphes similaires soient représentés par des points proches (Figure 4.1).

Cette technique est appelée aussi plongement explicite car la fonction de mise en correspondance est définie expressément.

Les techniques présentées dans la littérature peuvent être divisées en trois catégories principales :

- Méthodes basées sur la recherche de sous-structures ;
- Méthodes spectrales ;
- Méthodes basées sur la dissimilarité entre graphes.

La première catégorie se base sur l'idée d'avoir un dictionnaire de structures simples (motifs) et de représenter un graphe par un vecteur sur la base des occurrences des motifs dans le graphe. La clé de cette approche est la définition du dictionnaire qui doit être à la fois générique pour comprendre la plupart des motifs présents dans les données, et en même temps pas trop large pour que le vecteur résultant du plongement soit caractéristique des graphes à décrire. Un problème de ce type de technique est la prise en compte des attributs numériques associés aux sommets et aux arêtes. Dans cette catégorie, on peut citer les travaux de Nicolas Sidère et al. [Sidère et al. (2009), Sidère et al. (2008)] et l'article moins récent de [Inokuchi et al. (2000)].

La deuxième typologie, bien représentée par les travaux de Hancock et ces collaborateurs [Luo et al. (2003), Wilson et al. (2005), Ren et al. (2011)], est basée sur la théorie spectrale des graphes. La théorie spectrale des graphes concerne la compréhension de la façon dont les propriétés structurelles des graphes peuvent être caractérisées en utilisant les vecteurs propres de la matrice d'adjacence du graphe ou du Laplacien de cette

matrice. Généralement les techniques spectrales sont confrontées au problème de la gestion des attributs associés aux graphes.

Enfin, une dernière catégorie est celle représentée par un plongement basé sur le mapping des graphes dans l'espace de dissimilarité. Les graphes sont plongés en calculant la distance par rapport à un ensemble de graphes prototypes prédéfinis. Les distances qui en résultent peuvent être utilisées comme une représentation vectorielle du graphe considéré. Ces techniques peuvent facilement gérer n'importe quel type de graphe, orienté ou non orienté, avec ou sans attributs. Par contre le calcul de la distance entre les graphes est un problème NP-difficile comme on l'a montré dans le chapitre précédent, ce qui engendre des temps de calcul énormes pour produire les signatures. Le choix des graphes modèles est aussi un point sensible de cette méthode. Parmi les articles qui font partie de cette catégorie on peut citer [Bunke (2009), Jouili et Tabbone (2010)].

4.2 LE PLONGEMENT IMPLICITE : LES NOYAUX DE GRAPHES

Un noyau de graphe est une fonction qui associe à un couple de graphes un nombre réel, et qui possède des propriétés similaires à celles du produit scalaire défini sur les vecteurs. Plus formellement, si nous désignons par \mathbb{G} l'espace de tous les graphes, un noyau de graphe k est une fonction telle que :

$$k : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{R} \quad (4.1)$$

$$k(G_i, G_j) = k(G_j, G_i) \forall G_i, G_j \in \mathbb{G} \quad (4.2)$$

$$\sum_{i=1}^n \sum_{j=1}^n c_i \cdot c_j \cdot k(G_i, G_j) \geq 0 \quad \forall G_1, \dots, G_n \in \mathbb{G}, \forall c_1, \dots, c_n \in \mathbb{R} \quad (4.3)$$

L'éq.4.2 nécessite que la fonction k soit symétrique, tandis que l'éq.4.3 exige que k soit semi-définie positive.

De façon informelle, un noyau de graphe peut être considéré comme une mesure de similarité entre deux graphes. Toutefois, ses propriétés formelles, permettent de remplacer le produit scalaire de vecteurs par un noyau dans plusieurs algorithmes statistiques qui utilisent cet opérateur (et d'autres fonctions liées au produit scalaire, comme la norme euclidienne).

La technique des noyaux de graphes est ainsi dite plongement implicite parce que, comme pour le graph embedding, il y a une transformation des graphes dans un espace vectoriel. Par contre, avec les noyaux nous n'avons pas une définition explicite de cette transformation. Cette dernière est faite implicitement à travers la substitution du produit scalaire par le noyau de graphe défini.

Plusieurs noyaux de graphes ont été introduits dans la littérature, mais les noyaux de graphes construits sur la notion de sacs de motifs semblent être les plus prometteurs. La conception de ces noyaux, implique les choix suivants :

- la sélection d'un sac de motifs à partir d'un graphe ;

- la définition d'un noyau entre motifs ;
- la définition d'un noyau entre sacs basé sur le noyau entre motifs.

Le noyau de graphes correspond alors à la valeur du noyau entre les sacs de motifs.

Plusieurs noyaux ont été proposés sur la base de différents types de motifs : Vert [Mahé et Vert (2008)] propose de comparer l'ensemble des sous-arbres des deux graphes ; les noyaux de graphlets [Shervashidze et al. (2009)] sont basés sur le nombre de sous-graphes communs de deux graphes. Toutefois, la comparaison de motifs complexes nécessite des temps de calcul importants. En conséquence la plupart des noyaux de graphes sont basés sur des motifs plus simples tels que les *marches* [Kashima et al. (2003)], les *trails* [Dupé et Brun (2009)] ou les chemins. Nous définissons une marche comme une alternance de sommets et d'arêtes, un trail comme une marche avec arêtes distinctes et un chemin comme un trail avec sommets distincts. Les sacs de motifs peuvent être finis ou infinis. Les noyaux sur sacs infinis, comme les sacs de marches, sont calculés implicitement en utilisant des marches aléatoires définies sur les deux graphes à comparer [Kashima et al. (2003)]. Les noyaux basés sur les sacs (finis) de motifs peuvent être calculés explicitement en énumérant tous les motifs des graphes donnés [Dupas et Damiani (2009)]. Étant donné deux sacs de motifs, le noyau entre les deux sacs est souvent mesuré en utilisant une convolution [Haussler (1999)] des noyaux sur les motifs pour préserver la positivité du noyau.

Une autre approche dans la définition des noyaux de graphes est proposée par Neuhaus et Bunke [Neuhaus et Bunke (2007)]. Cette approche vise à définir les noyaux à partir de la distance d'édition de graphes. Le principal défi de cette approche est que la distance d'édition ne satisfait pas toutes les contraintes d'une métrique et donc ne conduit pas facilement à un noyau défini positif.

Ainsi, le défi, avec les méthodes basées sur les noyaux de graphe, est celui de définir des mesures de similarité qui :

- soient significatives, donnant des valeurs proches de 1 quand les graphes sont similaires et une valeur proche de 0 quand les graphes sont très différents ;
- prennent en compte les informations structurelles codées par les graphes ;
- soient rapides à calculer.

Ce défi, bien que de nombreuses propositions de noyaux aient été faites, reste encore ouvert. C'est dans ce cadre que nous avons conçu des nouvelles définitions de noyaux de graphes.

4.3 CONTRIBUTIONS AU PLONGEMENT DE GRAPHES

4.3.1 Des noyaux de graphes

Nous avons proposé deux différents noyaux de graphes, l'un basé sur la technique des sacs de motifs et l'autre basé sur la distance d'édition entre graphes. Avec ces deux noyaux nous avons essayé de surmonter les

difficultés présentes dans les propositions existantes, comme nous allons l'expliquer de suite.

La description d'un graphe par un sac de marches est pénalisée par un effet de "démarche chancelante" puisque une marche, même de longueur importante, peut passer toujours sur les mêmes arêtes décrivant ainsi juste une partie du graphe. Les sacs de trails réduisent considérablement cet effet et en plus ils peuvent capturer des caractéristiques non linéaires tels que des boucles qui ne sont pas décrites par les chemins.

Suard [Suard et al. (2007)] a proposé de nombreux noyaux basés sur les sacs de chemins. Notre proposition a été celle d'étendre le noyau moyen aux sacs de trails pour exploiter les avantages, cités ci-dessus, des trails par rapport aux chemins. En considérant un sac de trails et un noyau pour un couple de trails K_{trail} , nous proposons un noyau basé sur la convolution et appelé noyau moyen :

$$K_{mean}(T_1, T_2) = \frac{1}{|T_1|} \frac{1}{|T_2|} \sum_{t \in T_1} \sum_{t' \in T_2} K_{trail}(t, t'). \quad (4.4)$$

où T_1 et T_2 désignent deux sacs de trails.

Ce noyau est défini positif si et seulement si K_{trail} est défini positif sur le domaine des trails. Un noyau de trails $t = (v_1, e_1, \dots, v_n)$ et $t' = (v'_1, e'_1, \dots, v'_p)$ peut être dérivé à partir du noyau de chemins proposé par Kashima [Kashima et al. (2003)]. Le noyau vaut 0 si les deux trails n'ont pas la même longueur, autrement il est défini comme suit :

$$K_{trail}(t, t') = \delta(\varphi(v_1), \varphi(v'_1)) \prod_{i=2}^{|t|} \delta(\psi(e_{v_{i-1}v_i}), \psi(e_{v'_{i-1}v'_i})) \delta(\varphi(v_i), \varphi(v'_i)), \quad (4.5)$$

où $\varphi(v)$ et $\psi(e)$ désignent respectivement les étiquettes des sommets et des arêtes, tandis que δ est égal à 1 si les deux arguments sont égaux et 0 dans le cas contraire. Le noyau K_{trail} est défini positif puisqu'il est un produit de fonctions définies positives.

Un deuxième noyau proposé est basé sur la distance d'édition entre graphes, avec des modifications par rapport aux noyaux existants pour que le noyau soit défini positif. Soient $\{G_1, \dots, G_n\}$ un ensemble de graphes constituant les données de notre problème. Etant donné un noyau k , la matrice de Gram K associée aux données est une matrice $n \times n$ définie par $K_{i,j} = k(G_i, G_j)$. Comme montré par Steinke et Schölkopf (2008), l'inverse de K (ou son pseudo-inverse si K n'est pas inversible) pourrait être considéré comme un opérateur de régularisation d'un ensemble de vecteurs de dimension n . Inversement, l'inverse de n'importe quel opérateur de régularisation défini positif pourrait être considéré comme un noyau. Le schéma proposé est donc celui de construire un opérateur de régularisation sur les données et ensuite prendre son inverse pour obtenir un noyau. Comme opérateur de régularisation nous choisissons le Laplacien défini comme suit : étant donné un ensemble de graphes, $\{G_1, \dots, G_n\}$, on considère d'abord la matrice $W_{i,j} = e^{-\frac{d(G_i, G_j)}{\sigma}}$ où σ est un paramètre et $d(\cdot, \cdot)$ désigne la distance d'édition de graphes. Le Laplacien normalisé de $\{G_1, \dots, G_n\}$ est ensuite défini par $L = I - D^{-\frac{1}{2}} W D^{-\frac{1}{2}}$ où D est une matrice diagonale défini par $D_{i,i} = \sum_{j=1}^n W_{i,j}$. La théorie spectrale des graphes

| Méthode | FM | MM | FR | MR | Temps de calcul |
|-------------------------|-------|-------|-------|-------|-----------------|
| Appr. symbolique | 60.4% | 52.1% | 62.0% | 62.0% | ~ 2000 min |
| Noyau de trails | 61.2% | 50.4% | 62.8% | 65.6% | ~ 2 min |
| Noyau Laplacien | 59.2% | 55.2% | 57.7% | 60.9% | ~ 1 min 30 sec |

TABLE 4.1 – Précision et temps de calcul (de la phase d'apprentissage) sur la base de données PTC.

[Chung (1997)], affirme que L est une matrice symétrique, définie positive dont les valeurs propres appartient à l'intervalle $[0, 2]$. Malheureusement, le Laplacien est également bien connu pour être non inversible. De plus, le pseudo-inverse du Laplacien induit des instabilités numériques qui ne conduisent pas à un noyau fiable. Suivant Smola [Smola et Kondor (2003)], il faut régulariser la matrice L . La version régularisée, désignée par \tilde{L} , est définie de la manière suivante :

$$\tilde{L} = \sum_{i=1}^n r(\lambda_i) u_i u_i^t \quad (4.6)$$

où $\{\lambda_i, u_i\}$ est l'espace propre de L tandis que $r(\cdot)$ est une fonction de régularisation (nous avons utilisé $r(\lambda) = 1 + \eta\lambda$ où η est un paramètre).

Le Laplacien régularisé \tilde{L} est inversible et son inverse peut être pris comme noyau.

Nous avons testé les noyaux proposés et nous les avons comparés à une autre méthode d'apprentissage et classification, dans le cadre d'une application de biochimie : la prédiction de l'état cancérigène d'une molécule à partir de sa structure¹. Les composés chimiques sont séparés en 4 ensembles en fonction de ces réactions vis-à-vis : des souris femelles (FM), des souris mâles (MM), des rats femelles (FR) et des rats mâles (MR). Dans la Table 4.1, on montre les résultats de la précision de classification des noyaux proposés et de la méthode d'apprentissage basée sur l'appariement de graphes publié dans [Foggia et al. (2001a)]. Les temps de calcul de la phase d'apprentissage sont également indiqués. On peut donc noter que les méthodes basées sur les noyaux sont plus performantes que l'autre approche considérée, avec en plus une réduction très importante des temps de calcul.

4.3.2 Comparaison entre plongement implicite et explicite

Afin de bien comprendre le potentiel réel de la technique du plongement, et de comparer le plongement implicite et explicite nous avons mis en place une collaboration entre différents groupes de recherche qui travaillent sur les graphes et, plus spécifiquement, sur le plongement de graphes ; dans cette collaboration, dont j'ai été l'initiateur, nous avons convenu de l'application cible, des bases de données à utiliser et des conventions à utiliser pour rendre uniformes les expérimentations afin d'avoir des résultats comparables et analysables. Le premier résultat de

1. PTC challenge : <http://www.predictive-toxicology.org/>

cette collaboration a été la publication d'un article à la conférence Graph-Based Representation in Pattern Recognition (GbR) de 2013 [Conte et al. (2013b)], dont le contenu se résume comme suit.

Nous avons choisi d'appliquer les différentes techniques sélectionnées à la biochimie parce que dans ce domaine applicatif la représentation par graphe est plutôt naturelle et plusieurs travaux en ont montré l'efficacité. Nous avons mené des expériences sur quatre bases de données de molécules. Les molécules sont facilement converties en graphes en représentant les atomes par les sommets et les liaisons covalentes par les arêtes. Les sommets ont comme attributs le symbole de l'élément chimique et sur les arêtes la valence de la liaison est représentée :

AIDS² cette base de données se compose de deux classes (actif, inactif) de 2000 graphes représentant les molécules ayant une réactivité (ou pas) contre le VIH.

Mutagénicité³ cette base de données est divisée en deux classes concernant la mutagénicité (une des nombreuses propriétés défavorables d'un composé qui entrave sa capacité à devenir un médicament commercialisable) de 4337 molécules.

Predictive Toxicology Challenge (PTC)⁴ cette base de données traite de la cancérogénicité (positif ou négatif) des produits chimiques à l'aide des informations relatives à la structure chimique des molécules. Cette base de données contient environ 960 graphes.

Monoamine oxydase Dataset (MAO)⁵ ce problème est défini sur un ensemble de 68 molécules divisé en deux classes : les molécules qui inhibent la monoamine oxydase (des antidépresseurs) et celles qui ne le font pas.

Des algorithmes basés sur le plongement explicite

Parmi les techniques de plongement explicite, nous avons considéré les algorithmes suivants :

- **Plongement topologique (Topological Embedding [Sidère et al. (2009)])** La représentation vectorielle sera construite en comptant les occurrences de certains motifs contenus dans un dictionnaire. Chaque élément du vecteur correspond à la fréquence d'apparition d'un motif, ce qui représente un descripteur des différentes parties du graphe. Ensuite deux stratégies (discrétisation et clustering) sont utilisées pour inclure les attributs dans la représentation vectorielle (Figure 4.2).
- **Plongement multi-niveaux flou (Fuzzy Multilevel Graph Embedding [Luqman et al. (2013)])** Le plongement multi-niveaux flou effectue une analyse multi-niveaux du graphe pour extraire des informations à trois niveaux différents. Il s'agit notamment de l'information au niveau graphe (nombre de sommets, etc.), des informations au niveau structurel et les informations au niveau des attributs (Figure 4.3).

2. <http://www.iam.unibe.ch/fki/databases/iam-graph-database>

3. Voir note 2

4. <https://brun101.users.greyc.fr/CHEMISTRY/index.html>

5. Voir note 4

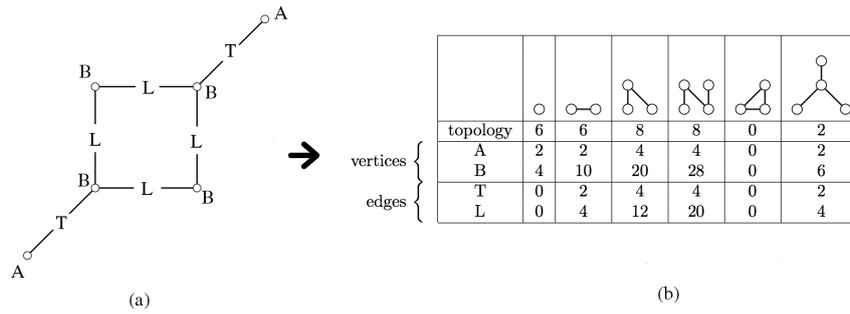


FIGURE 4.2 – Plongement topologique : les motifs (et leur occurrences) (b) correspondant à la représentation vectorielle du graphe in (a) ([Sidère et al. (2009)]).

| Graph order | Graph size | Embedding of node degree | Embedding(s) of subgraph(s) homogeneity | Embedding(s) of node attribute(s) | Embedding(s) of edge attribute(s) |
|-------------|------------|--------------------------|---|-----------------------------------|-----------------------------------|
|-------------|------------|--------------------------|---|-----------------------------------|-----------------------------------|

FIGURE 4.3 – La représentation vectorielle du plongement multi-niveau flou ([Luqman et al. (2013)]).

- **Plongement basé sur la statistique d'attributs (Attribute Statistics based Embedding [Gibert et al. (2012)])** Il consiste essentiellement dans le calcul de la fréquence d'apparition de certaines structures de sous-graphes très simples, tels que les sommets avec certaines étiquettes ou des structures sommet-arête-sommet avec des séquences d'étiquettes spécifiques.

Des algorithmes basés sur le plongement implicite

Parmi les techniques de plongement implicite, nous avons considéré les algorithmes suivants :

- **Noyau basé sur le Laplacien (Laplacian Kernel [Brun et al. (2011a)])** C'est le noyau que nous avons proposé, décrit dans la sous-section précédente.
- **Noyau basé sur les treelets (Treelet Kernel [Gauzère et al. (2012)])** Le noyau de Treelet est un noyau de graphes basé sur un sac de sous-graphes (nommés treelets) dans un graphe. Cette méthode énumère explicitement l'ensemble des treelets inclus dans un graphe. Puis le noyau entre deux graphes est défini comme une somme de noyaux entre les treelets communs des deux graphes.

Résultats et comparaison entre les méthodes

La Table 4.2 montre les taux de classification obtenus par les cinq méthodes considérées. En tenant compte des quatre ensembles de données, toutes les méthodes implicites et explicites semblent être compétitives et comparables. Bien sûr, en fonction des données, certaines variations peuvent apparaître, mais ces variations sont petites et ne peuvent pas être un critère pour choisir une méthode plutôt qu'une autre. Ainsi, d'autres considérations doivent être prises en compte.

Par exemple, la complexité de calcul ou la dépendance vis-à-vis des paramètres doivent être évaluées pour toutes ces approches. En plus, il est important de remarquer que la plupart des méthodes pourraient présenter

TABLE 4.2 – *Résultat de classification des différents méthodes de plongement sur les différentes bases de données.*

| | Mut. | AIDS | MAO | PTC | | | |
|-----------------------|------|------|------|------|------|------|------|
| | | | | FM | MM | FR | MR |
| Laplacian kernel | 70.2 | 92.6 | 90.0 | 59.2 | 55.2 | 57.7 | 60.9 |
| Treelet kernel | 77.1 | 99.1 | 91.2 | 58.7 | 61.9 | 60.4 | 60.8 |
| Topological Embedding | 77.2 | 99.4 | 91.2 | 65.9 | 67.5 | 68.7 | 63.7 |
| FMGE | 76.5 | 99.0 | 92.1 | 63.9 | 66.3 | 60.0 | 59.9 |
| Attribute Statistics | 76.5 | 99.6 | 90.6 | 64.8 | 63.1 | 67.9 | 59.7 |

certaines restrictions d'utilisation. Par exemple, les plongements explicites peuvent être favorisés par rapport à ceux implicites à chaque fois qu'une représentation vectorielle explicite est requise par exemple pour certains algorithmes qui nécessitent plus que le produit scalaire. D'autre part la plus part des noyaux peuvent être calculés entre deux graphes sans explorer au préalable la base de données utilisée dans l'application (apprentissage), alors que la plupart des méthodes explicites exigent l'ensemble des données pour calculer le plongement de graphes. Ainsi, les méthodes implicites peuvent être favorisées par rapport à celles explicites lorsque l'accès à l'ensemble des données est limité.

Bilan

La comparaison entre les deux techniques ne peut pas s'effectuer seulement en termes de performance sur des bases de données. Les méthodes présentent des caractéristiques spécifiques qui rendent une méthode plus efficace qu'une autre en fonction de l'application, des données disponibles, des contraintes de temps, etc. Le travail fait est donc intéressant, mais cela demeure seulement une première tentative de comparaison sur les différences entre les deux approches.

4.4 PERSPECTIVES

Une première perspective concernant mes travaux sur les noyaux de graphes est celle d'étendre l'utilisation des noyaux à d'autres domaines applicatifs. Grâce à la collaboration avec Luc Brun et Amal Mahboubi de l'ENSICAEN, qui continue encore aujourd'hui, nous avons déjà commencé à utiliser des noyaux et en développer d'autres dans le cadre d'un problème d'analyse vidéo [Mahboubi et al. (2013b;a; 2014), Conte et al. (2013b)], mais nous envisageons d'appliquer cette technique à d'autres problèmes similaires.

Nous envisageons aussi de continuer l'étude comparative entre les méthodes de plongement implicite et explicite en élargissant la collaboration avec d'autres groupes de recherche travaillant dans le plongement de graphes (comme par exemple le groupe de A. Tabbone du LORIA de Nancy) et en faisant des analyses plus approfondies des résultats pour comprendre le comportement des différentes méthodes dans différentes

applications. L'idée est celle de faire avancer cette étude sur la comparaison des méthodes de plongement de graphes pour révéler les points forts de ces méthodes en termes de capacité d'apprentissage, optimisation automatique des paramètres, complexité de calcul et d'autres critères intéressants. À notre avis, en effet, les deux approches ne doivent pas être considérées comme concurrentes ; au contraire, leur intégration pourrait fournir un outil puissant de RF.

Enfin, une dernière perspective est celle de définir un plongement pour d'autres structures de données que les graphes. Par exemple il serait intéressant de concevoir et utiliser des techniques de plongement de cartes généralisées dans les applications de classification ou apprentissage. Des premiers efforts ont été faits dans cette direction : dans [Gosselin et al. (2011)], il a été défini une signature d'une carte pour des applications d'indexation et classification. Mais d'autres possibilités restent encore ouvertes : par exemple la définition de noyaux de cartes qui pourraient permettre d'utiliser des outils d'apprentissage statistique (comme le SVM, la PCA, etc.) avec les cartes généralisés.

Troisième partie
L'analyse vidéo

INTRODUCTION AUX SYSTÈMES D'ANALYSE DE VIDÉOS

5

SOMMAIRE

| | |
|---|----|
| 5.1 ARCHITECTURE D'UN SYSTÈME DE VIDÉO SURVEILLANCE | 47 |
| 5.2 CONCLUSIONS | 49 |

LE domaine de l'analyse de vidéos attire de plus en plus l'intérêt des chercheurs au fur et à mesure qu'évoluent les technologies. La présence de caméras se généralise beaucoup, partout et dans tous les domaines. Les domaines de l'analyse vidéo les plus traités sont : l'annotation [Ozer et al. (1999)] et la recherche [Shearer et al. (2001)] dans les bases de données de vidéos, l'estimation du mouvement [Salotti et Laachfoubi (2001)] et la classification de vidéo [Hu et al. (2009)].

Un des domaines les plus développés ces dix dernières années a été la surveillance vidéo. La vidéo-surveillance d'environnements complexes devient de plus en plus intéressante, grâce à des améliorations significatives ces derniers temps dans le domaine du traitement de la vidéo en temps réel. D'ailleurs, il y a aussi un intérêt commercial croissant dans le développement de ce type de systèmes. Les systèmes de vidéo surveillance visent à l'observation en temps réel de personnes et/ou véhicules dans un certain environnement, conduisant à une description de leurs actions et interactions (voir par exemple [Darrell et al. (2000), Ferryman et al. (2000), Haritaoglu et al. (2000)]). De grands projets de recherche ont été consacrés à la vidéo surveillance aux États-Unis (par exemple, le projet de vidéo surveillance de la défense dans le cadre du projet DARPA [DARPA (1998)]), en Europe (voir les projets AVS-PV [Rota et Thonnat (2000)] et INDECT [Dziech et al. (2013)]) et au Japon (le projet Cooperative Distributed Vision [Matsuyama (1998)]).

Le domaine de la vidéo surveillance est devenu également un sujet de recherche très important comme le prouvent les dernières conférences [Ferryman (2013), Konrad et Domanski (2013)] et numéros spéciaux dans des revues [JIVP (2011), TIFS (2013)].

Depuis mes débuts dans le monde de la recherche, grâce à des contrats industriels, avec des entreprises italiennes, qui avaient pour objectif la réalisation de systèmes de vidéo surveillance intelligents mes activités de

recherche ont toujours traité de cette problématique. Beaucoup de mes contributions se situent donc dans ce domaine pour ce qui concerne les aspects applicatifs.

Bien que la plupart des problèmes liés à la vidéo surveillance aient été traités avec des techniques de RF statistique, il commence à paraître un certain nombre de méthodes qui se basent sur des représentations structurales [Paixao et al. (2008), Taj et al. (2006), Gomila et Meyer (2003), Loza et al. (2009), Aoun et al. (2014)]. Mon opinion est que le domaine structural peut fournir des solutions efficaces dans ce contexte puisque ce type de représentation peut, plus facilement qu'avec des vecteurs de caractéristiques, représenter les interactions complexes qui se produisent sur une scène et qui sont à la base de la compréhension d'une séquence vidéo. C'est pour cette raison que, dans la plupart des mes propositions, mes tentatives ont été d'utiliser les techniques de la RF structurale (décrites dans la partie précédente de ce mémoire) dans ce domaine.

Mes compétences en recherche sur la vidéo surveillance ont même abouti à la création, en collaboration avec d'autres chercheurs, d'une start-up académique qui vise à la mise en œuvre de systèmes pour la vidéo surveillance. Cela nous permet d'évaluer les problèmes réels liés à l'application pratique de notre recherche et nous oblige à élaborer des algorithmes robustes et fiables. D'autre part, nous pensons pouvoir, grâce à notre expertise acquise en recherche, réaliser des solutions réelles puissantes et efficaces.

De même c'est principalement dans ce domaine applicatif que je compte continuer ma recherche dans le futur puisque de nombreux problèmes de recherche restent encore ouverts.

Ce chapitre est dédié à la description de l'architecture générale d'un système de vidéo surveillance. On détaillera successivement chaque composante du système en décrivant l'état de l'art des techniques utilisées pour ensuite mieux démontrer l'intérêt de nos principales contributions.

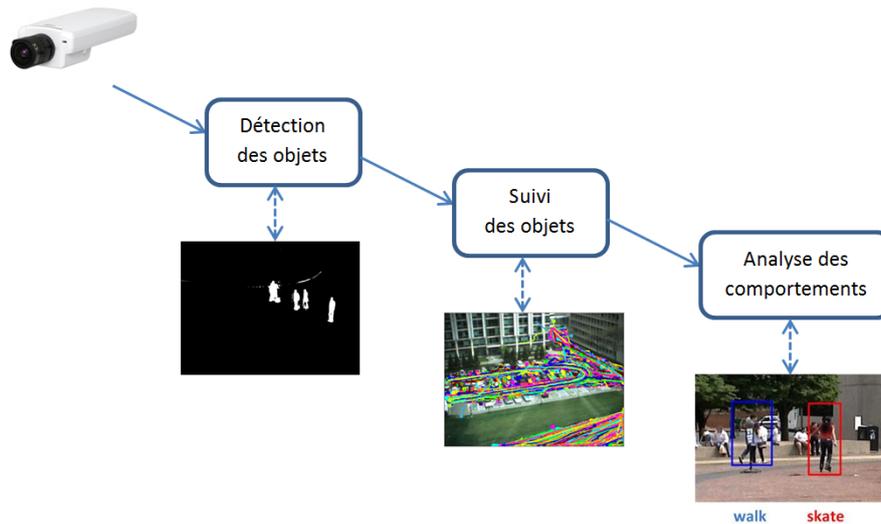


FIGURE 5.1 – L'architecture typique d'un système de vidéo surveillance.

5.1 ARCHITECTURE D'UN SYSTÈME DE VIDÉO SURVEILLANCE

L'architecture d'un système de vidéo surveillance est illustrée dans la Figure 5.1.

L'entrée du système est représentée par la vidéo sous forme de séquence de trames. Ensuite le système consiste en trois étapes principales : la *détection d'objets*, le *suivi d'objets* et l'*analyse comportementale*.

Détection d'objets

La première étape de ce système est la détection des objets en mouvement.

Le système commence par la segmentation des images du flux. Le but est de séparer l'image en deux ensembles de pixels : l'arrière-plan qui est l'ensemble de pixels dans l'image qui représente la partie fixe de la scène ; le premier plan est l'ensemble des pixels appartenant à des objets en mouvement. La sortie de ce premier module est un masque (nommé masque de premier plan ou "foreground mask") qui indique quels pixels appartiennent au premier plan (et donc quels pixels appartiennent à l'arrière-plan).

Ensuite il y a une phase d'analyse du premier plan qui est utilisée pour affiner le masque obtenu avec un premier traitement visant à éliminer tous les faux positifs, i.e. les pixels erronément détectés comme foreground (pixels de bruits, pixels d'objets qui ne sont pas d'intérêt pour l'application, etc.). Souvent ce traitement comprend des opérations morphologiques afin d'éliminer le bruit. Mais le traitement peut consister aussi en l'application d'algorithmes plus au moins complexes afin de détecter correctement les seuls pixels d'intérêt.

Suit ensuite une phase de détection des composantes connexes dans les pixels de foreground pour détecter les régions qui représentent les objets en mouvement.

La sortie de cette étape est l'ensemble des objets en mouvement, ces

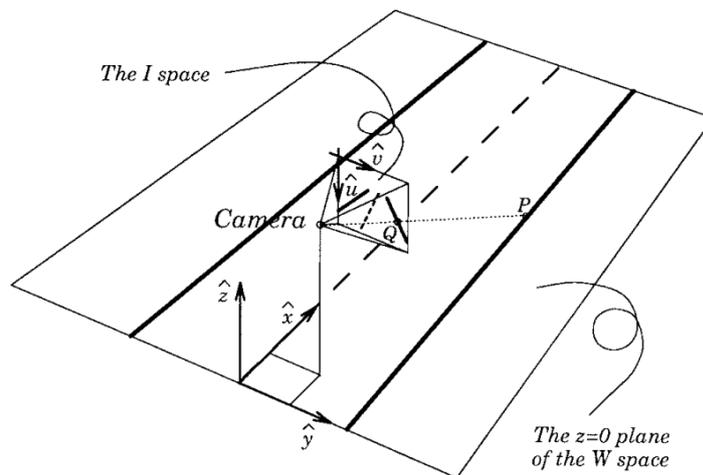


FIGURE 5.2 – Relation entre le plan image et le monde réel (M.Bertozzi et A.Broggi (1998)).

derniers typiquement représentés par les boîtes englobantes des régions connexes du masque de foreground.

Suivi d'objets

La deuxième étape du système prend en entrée les objets en mouvement et, pour chacun des objets en mouvement, calcule la position dans la scène et l'ajoute à la liste des positions de l'objet dans les trames précédentes. Par conséquent, la sortie de cette étape correspond aux trajectoires des objets. Un des problèmes les plus difficiles dans cette étape est lié aux occlusions, c'est-à-dire quand un objet d'intérêt est partiellement ou totalement couvert par un autre objet de la scène. Ce problème est dû au fait que la compréhension de la scène doit être faite en trois dimensions alors que le traitement opère sur des images bi-dimensionnelles.

Analyse comportementale

Les positions listées dans les trajectoires sont exprimées en termes de pixels sur l'image. Pour analyser les événements sur la scène, par contre, il faut les positions des objets en termes de coordonnées dans le monde réel (par exemple la distance en mètres par rapport à la caméra).

Il faut alors un module pour la transformation des coordonnées de pixels en coordonnées dans le monde réel à l'aide de certains paramètres de calibration de la caméra. Le module de "inverse perspective mapping" (IPM) est consacré à cette tâche [M.Bertozzi et A.Broggi (1998)]. Le principe de l'IPM est le suivant. Commençons par définir deux espaces euclidiens :

- $W = \{(x, y, z)\} \in E^3$ qui représente l'espace 3D du monde réel ;
- $I = \{(u, v)\} \in E^2$ qui représente l'espace 2D de l'image ;

L'image acquise par la caméra appartient à l'espace à deux dimensions I , tandis que les coordonnées transformées par le IPM sont définies dans W avec $z = 0$. La relation entre W et I est illustrée dans la Figure 5.2. Pour déterminer les coordonnées réelles, l'IPM a besoin des paramètres

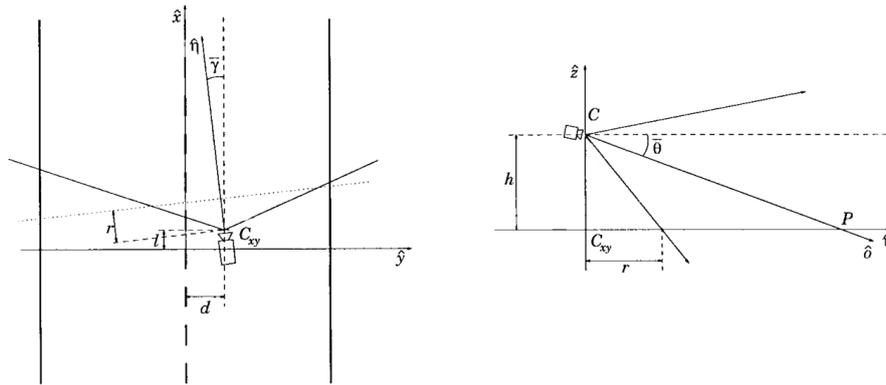


FIGURE 5.3 – Les paramètres nécessaires à l'IPM (M.Bertozzi et A.Broggi (1998)).

suivants (Figure 5.3) :

- Le point d'observation : la position de la caméra $C = (l, d, h) \in W$;
- L'angle d'observation : la position de l'axe optique définie par les deux angles :
 - γ_c : l'angle formé par la projection de l'axe optique sur le plan $z = 0$ et l'axe x ;
 - θ_c : l'angle formé par la projection de l'axe optique sur le plan $x = 0$ et l'axe z ;
- L'ouverture de l'objectif 2α ;
- La résolution de la caméra $N \times N$.

Avec ces paramètres la transformation $I \rightarrow W$ est définie par :

$$\begin{cases} x(u, v) = \frac{h}{\tan[\theta_c - \alpha] + u \frac{2\alpha}{N-1}} \cdot \cos [(\gamma_c - \alpha) + v \frac{2\alpha}{N-1}] + l \\ y(u, v) = \frac{h}{\tan[\theta_c - \alpha] + u \frac{2\alpha}{N-1}} \cdot \sin [(\gamma_c - \alpha) + v \frac{2\alpha}{N-1}] + d \\ z(u, v) = 0 \end{cases}$$

Les trajectoires des objets sont ensuite analysées par l'étape d'"analyse comportementale" pour déterminer les événements qui se produisent dans la scène. Cette étape est très dépendante de l'application choisie.

5.2 CONCLUSIONS

Dans ce chapitre nous avons présenté l'architecture d'un système de vidéo surveillance. Le système dans son ensemble pose beaucoup de défis : en effet chaque composante décrite est caractérisée par ses problèmes propres. Pour cette raison dans la littérature, il n'y a pas d'auteurs qui affrontent le framework du début à la fin, mais la tendance est de se concentrer sur une seule composante (par exemple la détection ou le suivi).

Dans mes recherches, j'ai aussi abordé les différentes parties individuellement. En conséquence je présenterai dans les prochains chapitres une composante à la fois, en détaillant l'état de l'art et mes contributions.

DÉTECTION D'OBJETS

6

SOMMAIRE

| | | |
|-----|-------------------------|----|
| 6.1 | ÉTAT DE L'ART | 53 |
| 6.2 | CONTRIBUTIONS | 55 |
| 6.3 | PERSPECTIVES | 73 |

LA détection d'objets est le processus consistant à séparer les objets en mouvement de l'arrière-plan de la scène.

La phase de détection d'objets consiste généralement en deux traitements :

- Le traitement au niveau pixel : segmentation entre l'ensemble des pixels de l'arrière plan et de l'ensemble des pixels de premier plan (foreground).
- Le traitement au niveau région : regroupement des pixels en régions à travers certaines règles de similarité (typiquement l'adjacence entre pixels).

La détection d'objets n'est pas un problème trivial pour de nombreuses raisons. Tout d'abord, le problème de déterminer quels sont les objets à détecter est très dépendant de l'application. Par exemple les personnes sont des objets d'intérêt dans les applications de vidéo surveillance, alors que dans une application de monitoring du trafic sur les autoroutes ce sont les voitures qui font l'objet de la détection. Il n'existe aucun algorithme qui fournisse une solution générale pour tous types d'applications. En outre, il y a beaucoup de problèmes liés à la variabilité de la scène : la variation de taille des objets, la variabilité de la couleur, de la luminosité de la scène, etc.

Après un état de l'art des techniques traitant la détection d'objets (Section 6.1), nous allons présenter nos contributions (Section 6.2) dont, certaines d'entre elles, sont basées sur une représentation structurelle de l'information. En particulier nous venons de publier (Conte et Damiand

(2014)) une méthode pour la réduction du bruit dans les masques de foreground qui est basée sur les cartes combinatoires.

La Section 6.3 est dédiée aux perspectives envisagées dans le contexte de la détection d'objets.

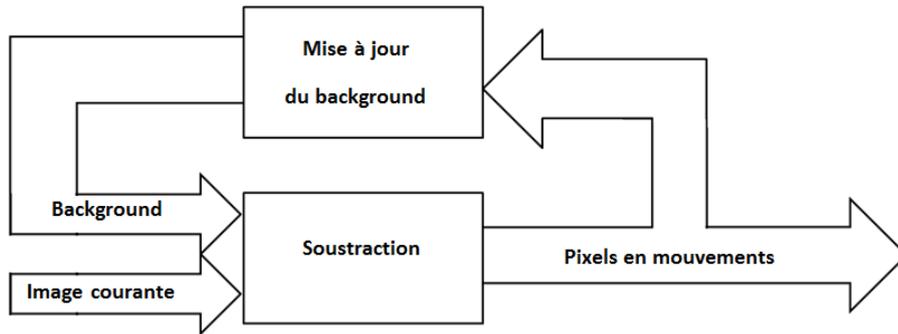


FIGURE 6.1 – Le schéma générale de la technique de soustraction de fond.

6.1 ÉTAT DE L'ART

Les techniques de détection d'objets utilisent principalement les trois approches suivantes :

- La différence temporelle ;
- Le flux optique ;
- La technique de soustraction de fond (ou *background subtraction*).

La première catégorie se base sur la détection des pixels en mouvements à travers la différence entre trames adjacentes. Cette différence peut être effectuée entre deux images temporellement adjacentes (*single difference*) ou entre trois images adjacentes (*double difference*). Cette technique a été la première à être proposée en 1979 [Jain et Nage (1979)] et puis reprise par un des articles les plus représentatifs de la vidéo surveillance [Haritaoglu et al. (2000)].

Le travail pionnier du flux optique est sûrement celui de Horn et Schunck (1981) où le concept a été présenté pour la première fois. Le flux optique est une approximation du mouvement de l'image basée sur les dérivés locales dans une séquence d'images donnée. Les parties en mouvement provoquent des variations temporelles de la luminosité de l'image. Il est donc supposé que tous les changements d'intensité sont dus au mouvement seulement. La sortie du flux optique est le vecteur de mouvement pour chaque pixel de l'images. Ces vecteurs sont ensuite utilisés pour la segmentation de l'image. Une variante du calcul du flux optique, surtout dans le but de réduire les temps de calcul, est la détermination du flux optique par blocs de pixels [Kim et Hwang (1999)].

La méthodologie de loin la plus utilisée est celle de la soustraction du fond. L'idée principale est celle d'avoir une image de référence (appelé *background*) qui représente la scène sans aucun objet en mouvement et ensuite de calculer la différence entre la trame courante et le background. A cause des variations de la scène pendant la séquence vidéo, à la différence entre image courante et background s'ajoute une phase de mise à jour de l'image de référence. Le schéma de la technique de soustraction de fond est présenté dans la Figure 6.1.

Les différents algorithmes de l'état de l'art se distinguent sur deux points principaux : comment le background est représenté et comment il est mis à jour le long de la séquence. Les pixels du background peuvent être représentés soit par une seule valeur [Marcenaro et al. (2002)] ou par une distribution probabiliste. Un simple modèle statistique est utilisé à travers l'introduction d'une distribution Gaussienne pour les pixels du background (l'article de référence étant [Wren et al. (1997)]). Une variante plus complexe, pour prendre en compte la variabilité du background pendant la séquence vidéo, est la représentation des pixels par un mélange de Gaussiennes ; ce modèle, connu sur le nom anglais de MOG (Mixture of Gaussians), est décrit par Stauffer et Grimson (2000) pour la première fois et il est l'un des plus utilisés pour la détection du foreground. Une autre approche possible pour la représentation de l'arrière-plan est une description par les contours des objets [Cavallaro et Ebrahimi (2001)] : cette solution est moins sensible aux changements de luminosité, mais augmente les temps de calcul à cause de l'algorithme de détection des contours.

Une fois le type de représentation du background choisi, ce dernier est normalement mis à jour toutes les trames pour suivre les éventuels changements dans la scène (variation de luminosité, déplacement d'objets qui appartenait à la partie statique de la scène, etc.). La mise à jour peut être non récursive [Toyama et al. (1999)] ou récursive (voir les articles de référence déjà cités [Haritaoglu et al. (2000), Wren et al. (1997)]). Les techniques de la première catégorie calculent leur mise à jour sur la base des dernières trames à travers une prédiction de la valeur des pixels de background. La deuxième technique met à jour le modèle de l'arrière-plan en utilisant la trame courante et le background courant.

Bilan de l'état de l'art

Les techniques basées sur la différence temporelle posent problème pour la détection d'objets avec des couleurs uniformes dans leur apparence (problème appelé *foreground aperture*) ; dans ce cas en effet la différence entre deux trames successives fournit des valeurs nulles pour les pixels des objets en mouvement. En plus, ces techniques dépendent fortement de la vitesse des objets (ou pareillement du nombre d'images par seconde de la vidéo). Les techniques basées sur le flux optique souffrent aussi du problème de la *foreground aperture*. En outre, ces techniques sont relativement coûteuses en terme de temps, même dans la version par bloc ; par contre elles sont beaucoup employées dans l'analyse de vidéo avec caméra mobile (où les techniques de soustraction de fond ne peuvent pas marcher). Les techniques de soustraction de fond sont les plus prometteuses et les plus utilisées. Néanmoins elles aussi sont gourmandes en temps de calcul et en plus ne sont pas fiables dans tous les scénarios possibles. En effet, on ne peut pas trouver une solution générale et rapide pour toutes les situations : un compromis entre la rapidité de calcul et la robustesse de la solution doit être trouvé. Nous l'avons montré dans un article de comparaison des principales techniques de soustraction du fond [Conte et al. (2010c)].

Pour cette raison, notre idée a été celle d'utiliser un algorithme basique (et donc rapide) de soustraction de fond en concevant et en ajoutant

des post-traitements qui éliminent les problèmes de l’algorithme de base, augmentant la qualité de la solution finale, sans pour autant faire croître beaucoup les temps de calcul.

Les différents post-traitements proposés sont décrits, par ordre de parution, dans la section suivante. Elles correspondent à : la détection du camouflage et reconstruction des objets [Conte et al. (2010a)], la suppression des reflets [Conte et al. (2012b)], le dé-bruitage des masques de foreground à l’aide des cartes topologiques [Conte et Damiand (2014)].

6.2 CONTRIBUTIONS

Détection du camouflage et reconstruction des objets

Le camouflage est un problème qui survient lorsque les caractéristiques des pixels d’un objet de premier plan sont trop similaires à l’arrière-plan. L’effet est que la différence de ces pixels avec le background est en dessous du seuil, et par conséquent ces pixels sont considérés à tort comme des pixels de background. Le problème du camouflage fait que le masque de foreground d’un objet est divisé en deux ou plusieurs régions qui peuvent se positionner en beaucoup de configurations différentes (comme on peut le voir dans la Figure 6.2.



FIGURE 6.2 – Des exemples de configurations des régions d’un objet à cause du problème du camouflage.

Le problème de la récupération de ce genre d’erreurs est équivalent à définir comment plusieurs régions du premier plan peuvent être regroupées afin de représenter convenablement l’objet d’intérêt (dans notre cas une personne). En examinant une seule région, il n’est pas possible de déterminer si elle peut être considérée comme une partie d’un objet plus grand, ou si elle est faussement générée lors du processus de détection du foreground. Donc, pour reconstruire les objets nous avons besoin de définir un modèle de l’objet complet et une procédure pour regrouper convenablement les régions, de manière à s’adapter adéquatement au modèle.

Dans [Conte et al. (2010a)] nous nous sommes concentrés sur la détection de personnes, mais l’idée sous-jacente peut être généralisée à la détection d’autres types d’objets (comme les voitures, les animaux, etc).

Il est important de remarquer que le modèle ne peut pas être exprimé en pixels. La perspective provoque, en effet, qu'une même configuration de pixels peut représenter un objet de taille réelle différente en fonction de la distance de la caméra. Par conséquent, le modèle de recombinaison doit être défini en termes de taille réelle des objets à reconstruire, et la procédure de l'IPM (vu dans le chapitre précédent) doit être appliquée avant l'algorithme de regroupement.

Le modèle que nous proposons représente une personne par une boîte englobante définie par quatre paramètres h_1 , h_2 , b_1 et b_2 qui définissent, respectivement, la hauteur minimale et maximale réelle et la largeur minimale et maximale réelle. Le pseudo-code de l'algorithme est présenté dans Alg. 1. Le but de l'algorithme est de regrouper deux ou plusieurs régions pour former une région unique représentant une personne (selon le modèle défini). En cohérence avec la représentation adoptée pour une personne, l'algorithme représente chaque région par sa boîte englobante. La procédure consiste à fusionner itérativement des couples de régions jusqu'à ce que la nouvelle région corresponde le mieux au modèle défini. Deux régions sont regroupées si et seulement si toutes les conditions suivantes sont vérifiées :

- R1. La projection sur l'axe horizontal des deux régions considérées se superposent (voir la Figure 6.3). On notera que les coordonnées des deux boîtes sont exprimées en pixels.
- R2. La hauteur réelle (en mètres) de la région résultant après la fusion est comprise entre h_1 et h_2 .
- R3. La largeur réelle de la région résultant après la fusion est comprise entre b_1 et b_2 .

Pour vérifier les deux dernières conditions, nous construisons d'abord la boîte qui regroupe les boîtes englobantes des régions considérées et ensuite on détermine sa taille réelle par l'IPM.

Il est important de souligner que la méthode proposée n'est pas coût-

Algorithme 1 : Le pseudo-code de l'algorithme de regroupement.

```

S ← toutes les régions détectées
C ← S × S
while ∃(X, Y) ∈ C do
  R1 ← droitep(X) ≥ gauchep(Y) ∧ gauchep(X) ≤ droitep(Y)
  Z ← X ∪ Y
  <calcul de la taille réelle de Z>
  R2 ← hauteurr(Z) ∈ [h1, h2]
  R3 ← largeurr(Z) ∈ [b1, b2]
  if R1 ∧ R2 ∧ R3 then
    <fusionner les deux régions X et Y >
    S ← S - {X, Y}
    S ← S ∪ {Z}
    C ← S × S
  else
    C ← C - {(X, Y)}

```

teuse en calcul, car le nombre de régions détectées dans une image ne dépasse presque jamais la dizaine.

La validation expérimentale de la méthode proposée a été effectuée à travers l'évaluation de l'amélioration des performances après son application en tant que post-traitement de la sortie de quatre algorithmes bien connus de soustraction du background. Les algorithmes suivants ont été considérés :

- *Mixture of Gaussians (MOG)*, dans la version proposée par Kaewtrakulpong et Bowden (2001) ;
- *Enhanced Background Subtraction (EBS)*, proposé par nous-même dans Conte et al. (2005c) ;
- *Self-Organizing Background Subtraction (SOBS)*, proposé par Maddalena et Petrosino (2008) ;
- *Statistical Background Algorithm (SBA)*, proposé par Li et al. (2003).

La performance est mesurée en utilisant l'indice *f-score*, défini comme la moyenne harmonique de *précision* et *rappel*. Les deux derniers indices ont été calculés selon les définitions suivantes pour les vrais positifs (TP), les faux positifs (FP) et les faux négatifs (FN) :

$$TP = \sum_{g \in G} \sum_{d \in D} \frac{|g \cap d|}{|g \cup d|} \quad FP = \sum_{d \in D} \frac{|d| - \max_{g \in G} |d \cap g|}{|d|}$$

$$FN = \sum_{g \in G} \frac{|g| - \max_{d \in D} |d \cap g|}{|g|}$$

où G et D sont respectivement l'ensemble des objets de la vérité terrain et l'ensemble des objets détectés par l'algorithme (les objets sont représentés par leur boîte englobante) ; avec \cap et \cup on désigne l'intersection et l'union entre deux boîtes, tandis que $|\cdot|$ est utilisé pour indiquer la surface d'une région.

Les tests ont été effectués sur un ensemble de cinq séquences vidéo réelles enregistrées en intérieur et en extérieur.

Les trois vidéos NA1-NA3 ont été acquises par nous-même et présentent différentes conditions d'éclairage et météorologiques, avec plusieurs situations complexes. La vidéo PETS appartient à l'ensemble de

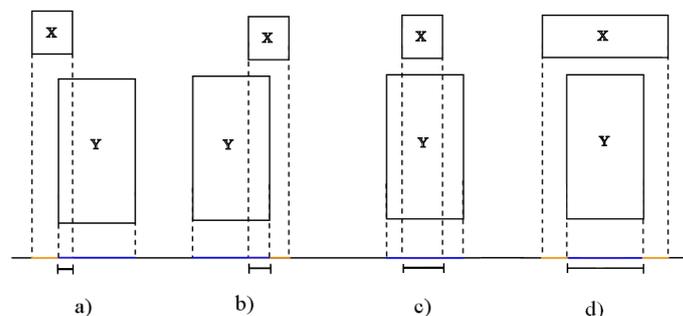


FIGURE 6.3 – Les possibles configurations de superposition entre les projections de deux boîtes sur l'axe horizontal pour que les deux régions soient fusionnées.

| Vidéo ID | Type d'objet | EBS | | | MOG | | | SBA | | | SOBS | | |
|----------|--------------|-----|-----|-------|-----|-----|--------|-----|-----|-------|------|-----|-------|
| | | av. | ap. | ame. | av. | ap. | ame. | av. | ap. | ame. | av. | ap. | ame. |
| NA1 | tous | .67 | .67 | 0.3% | .66 | .67 | 2.0% | .22 | .30 | 38.5% | .53 | .57 | 8.0% |
| | pers. | .73 | .75 | 2.1% | .65 | .64 | -0.1% | .21 | .31 | 46.9% | .52 | .59 | 14.6% |
| NA2 | tous | .77 | .87 | 12.9% | .67 | .68 | 0.7% | .26 | .28 | 8.5% | .43 | .51 | 17.4% |
| | pers. | .73 | .87 | 19.8% | .65 | .66 | 2.3% | .20 | .27 | 35.9% | .42 | .45 | 7.9% |
| NA3 | tous | .55 | .69 | 24.7% | .64 | .65 | 0.7% | .20 | .23 | 10.9% | .29 | .42 | 46.4% |
| | pers. | .55 | .70 | 27.1% | .65 | .66 | 1.5% | .20 | .23 | 13.3% | .30 | .43 | 41.1% |
| PETS | tous | .77 | .80 | 3.6% | .75 | .65 | -13.4% | .64 | .63 | -1.5% | .58 | .61 | 4.0% |
| | pers. | .72 | .82 | 13.0% | .62 | .64 | 3.2% | .42 | .42 | 0.1% | .54 | .60 | 9.8% |
| MSA | tous | .85 | .90 | 6.6% | .52 | .54 | 3.5% | .16 | .21 | 26.4% | .85 | .85 | - |
| | pers. | .84 | .92 | 9.7% | .54 | .56 | 4.3% | .61 | .62 | 1.5% | .82 | .83 | 1.5% |

TABLE 6.1 – Valeurs du *f*-score avant et après l'application de la méthode proposée pour les quatre algorithmes considérés. Pour chaque vidéo et chaque algorithme le pourcentage d'amélioration est montré.

données publiées lors de l'édition 2006 de la conférence PETS dédiée expressément à l'évaluation des systèmes de vidéo surveillance. La séquence MSA a été présentée dans [Maddalena et Petrosino (2008)], et montre une scène en intérieur.

La Table 6.1 rapporte la performance des algorithmes considérés avant et après l'application de la méthode de récupération des erreurs de camouflage, ainsi que le pourcentage d'amélioration. Les résultats sont donnés dans deux cas : dans le premier cas tous les objets de la scène ont été considérés et dans le deuxième, seulement les personnes sont prises en compte. Comme on a utilisé un modèle pour les personnes, dans le deuxième cas on remarque des pourcentages d'amélioration plus élevés. Mais il est possible de noter que, dans la grande majorité des cas, l'utilisation de la méthode proposée améliore les performances de détection d'objets.

Pour conclure, je voudrais proposer une considération liée à la méthode proposée : le modèle utilisé dans ces travaux était assez simpliste puisque les régions sont regroupées sans exploiter aucune connaissance de sa nature (région de bruit, région d'ombre, etc.). Ce modèle simple permet un traitement rapide de l'image, néanmoins quelques informations supplémentaires pourraient être utilisées (comme la forme, la couleur, etc.) pour améliorer encore les résultats obtenus.

Suppression des reflets

Dans le contexte de la détection d'objets, la suppression des ombres et des reflets est une tâche importante. En fait, si une ombre ou un reflet est inclus par erreur dans la boîte englobante d'un objet, plusieurs problèmes peuvent surgir : parmi eux, des erreurs dans l'estimation de la taille réelle ou de la forme de l'objet, de sa position, etc. De plus, les ombres peuvent, de manière erronée, causer la fusion de plusieurs objets en une seule région (Figure 6.4).

Alors que de nombreux articles sont consacrés à la suppression des ombres (voir l'article de synthèse de Prati et al. (2003)), le problème des reflets a reçu beaucoup moins d'attention. Cependant, dans certains environnements, des reflets peuvent être plus probables que des ombres, et généralement ils sont plus difficiles à traiter.

D'ailleurs les reflets ne peuvent pas être traités de la même manière



FIGURE 6.4 – Un effet indésirable de l'ombre : deux objets fusionnée dans une seule région.

que les ombres à cause de ses propriétés spécifiques. La position d'une ombre dépend des sources de lumière, tandis que les reflets (en supposant que la surface réfléchissante soit le sol) sont toujours situés sous l'objet correspondant. Les pixels des reflets sont plus similaires à l'arrière-plan que les pixels appartenant aux objets, alors que les pixels appartenant aux ombres sont bien différents du background. Cependant les pixels des reflets ne sont pas souvent suffisamment similaires à l'arrière plan pour être considérés comme fond par l'algorithme de background subtraction : cela se produit parce qu'une partie de la couleur du sol se mélange avec la couleur de l'objet réfléchi pour former la couleur du reflet. Donc le rendu visuel d'un reflet est très différent de ce des ombres.

Nous avons mis au point une méthode exploitant les propriétés que l'on vient de décrire pour détecter et supprimer les reflets.

Chaque boîte englobante, avec les pixels de la région représentée, constitue l'entrée de l'algorithme de suppression des reflets. Le but de l'algorithme est de détecter la ligne horizontale, ci-après dénommée la ligne de coupe, qui sépare l'objet (formé par les pixels au-dessus de la ligne) de son reflet (formé par les pixels en dessous de la ligne). Une fois la ligne de coupe détectée, la suppression du reflet consiste à supprimer du masque de foreground les pixels en dessous de cette ligne.

Sur la base des propriétés des reflets, la ligne de coupe idéale est déterminée de sorte que les deux conditions suivantes soient satisfaites (voir la Figure 6.5) :

- pour la région en dessous de la ligne de coupe, la moyenne des différences entre la couleur des pixels de l'objet détecté et la couleur des pixels correspondants dans l'arrière-plan est minimale ;
- pour la région au-dessus de la ligne de coupe, la moyenne des différences entre la couleur des pixels de l'objet détecté et la couleur des pixels correspondants dans l'arrière-plan est maximale.

En d'autres termes l'on recherche la ligne pour laquelle l'apparence de la région en dessous de la ligne est plus similaire à l'arrière plan que l'apparence de la région qui se trouve au-dessus de cette ligne.

Naturellement, afin de déterminer la ligne de coupe optimale, il est plus facile de réduire ces deux critères à une seule condition. Une méthode couramment utilisée dans l'optimisation multi-objectif est la somme pondérée des différentes fonctions objectives.

Dans notre cas (voir la Figure 6.6 pour un exemple visuel de la notation utilisée), si l'on désigne par D_y^a la moyenne des différences entre la couleur des pixels de l'objet détecté et la couleur des pixels correspondants dans

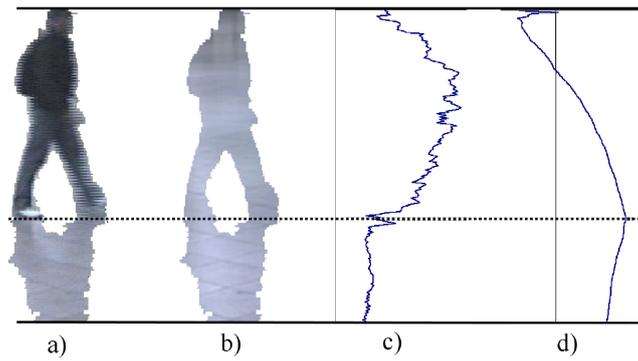


FIGURE 6.5 – Un exemple d'objet avec reflet et les conditions pour la détection de la ligne de coupe. a) La région de foreground et b) le background correspondant. c) La fonction d_y pour chaque ligne de l'image (y est l'axe vertical, la valeur de d_y se situe sur l'axe horizontal). d) La fonction Δ_y pour chaque ligne de l'image.

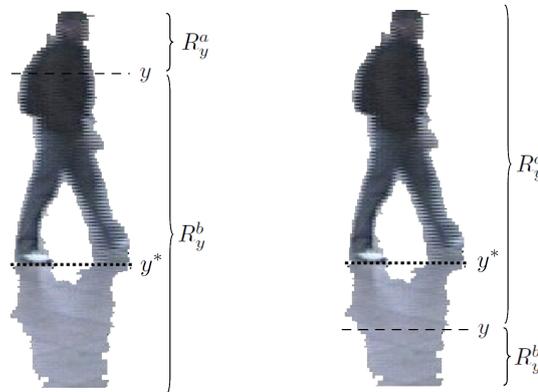


FIGURE 6.6 – L'illustration visuelle de la notation utilisée.

l'arrière-plan pour la région au dessus de la ligne y , et par D_y^b la même mesure pour la région en dessous de y , alors nous pouvons définir une fonction objectif Δ_y à maximiser :

$$\Delta_y = c^a \cdot D_y^a + c^b \cdot D_y^b \quad (6.1)$$

où $c^a > 0$ et $c^b < 0$, parce que nous voulons maximiser D^a et minimiser D^b . Ces coefficients peuvent être utilisés pour attribuer un poids différent aux deux critères, si c'était nécessaire dans un contexte applicatif spécifique. Puisque dans notre cas, nous n'avons pas de raison *a priori* pour préférer l'un sur l'autre, nous pouvons définir $c^a = +1$ et $c^b = -1$, réduisant ainsi notre problème à la maximisation de $D_y^a - D_y^b$.

Afin d'évaluer quantitativement la différence de couleur, nous introduisons les notations suivantes : $F(x, y)$ est la couleur du pixel à la position (x, y) dans la région de foreground, $B(x, y)$ la couleur du pixel de background à la même position, et r_y l'ensemble des pixels appartenant à la ligne générique y de la région de foreground. Nous mesurons la différence moyenne de la couleur, le long de la ligne y de l'objet comme suit :

$$d_y = \frac{\sum_{(x,y) \in r_y} \|F(x,y) - B(x,y)\|}{|r_y|} \quad (6.2)$$

où $\|\cdot\|$ est la norme Euclidienne dans l'espace de couleur RGB, et $|\cdot|$ indique la cardinalité d'un ensemble.

Etant donnée la définition de d_y on peut maintenant exprimer D_y^a et D_y^b comme suit :

$$D_y^a = \frac{1}{|R_y^a|} \cdot \sum_{i \in R_y^a} d_i \quad D_y^b = \frac{1}{|R_y^b|} \cdot \sum_{i \in R_y^b} d_i \quad (6.3)$$

où R_y^a et R_y^b sont les ensembles des lignes au dessus et en dessous de y , respectivement.

Enfin, nous pouvons exprimer la fonction objectif Δ_y par :

$$\Delta_y = D_y^a - D_y^b = \frac{1}{|R_y^a|} \cdot \sum_{i \in R_y^a} d_i - \frac{1}{|R_y^b|} \cdot \sum_{j \in R_y^b} d_j \quad (6.4)$$

La Figure 6.5d montre le graphique de la fonction Δ_y pour toutes les lignes de la région considérée.

La ligne de coupe idéale y^* peut être déterminée par la recherche du maximum de Δ_y .

Dans les cas réels, la fonction Δ_y montre quelques maxima parasites en plus de celui correspondant à la ligne de coupe idéale. Ces maxima sont dus à l'effet du bruit et de la non uniformité dans la couleur de l'objet de premier plan, qui peut être localement très similaire à la couleur de fond. Nous avons donc ajouté, dans la recherche du maximum, des critères pour ne pas tomber sur des maxima locaux. Un maximum n'est pas considéré si :

- il est trop isolé ; la raison d'être de ce critère est qu'un maximum isolé est probablement du bruit. Un maximum local à la ligne \hat{y} est considéré comme isolé si la moyenne de Δ_y sur une fenêtre centrée sur \hat{y} est inférieure à une fraction fixée de la valeur de $\Delta_{\hat{y}}$, c'est-à-dire :

$$\frac{1}{2w} \sum_{\substack{i=\hat{y}-w \\ i \neq \hat{y}}}^{\hat{y}+w} \Delta_i \leq \tau \Delta_{\hat{y}} \quad (6.5)$$

où $2w$ est la largeur de la fenêtre de recherche centrée sur \hat{y} et τ est un paramètre de l'algorithme dont la valeur est comprise entre 0 et 1 ;

- la position du maximum est sur une ligne au-dessus de la moitié de la région en examen ; il est géométriquement peu probable qu'un reflet soit plus grand que l'objet réel, si la surface du sol est horizontale et l'objet n'est pas sensiblement incliné ;
- la valeur du maximum est négative ; une valeur négative de Δ_y signifierait que l'apparence de l'objet est plus proche des valeurs de background que l'apparence de sa réflexion, ce qui est incompatible avec la nature du phénomène du reflet.

Algorithme 2 : Algorithme de recherche de la ligne de coupe idéal.

```

Sum ← 0
for i = 0 to height - 1 do
  di ←  $\sum_{(x,y) \in r_i} \|F(x,y) - B(x,y)\| / |r_i|$ 
  Sum ← Sum + di

SumAbove ← d0
SumBelow ← Sum - d0
for y = 1 to height - 1 do
  Δy ← SumAbove/y - SumBelow/(height - y)
  SumAbove ← SumAbove + dy
  SumBelow ← SumBelow - dy

BestMax ← -1
BestCut ← -1
for y = height/2 to height - 1 do
  if Δy is a local maximum AND Δy > 0 AND Δy is not isolated then
    if Δy > BestMax then
      BestMax ← Δy
      BestCut ← y

if BestMax > 0 then
  return BestCut
else
  return Nothing

```

L'algorithme que nous avons mis en œuvre est fourni dans Algorithme 2.

Par souci de comparabilité, la validation expérimentale de la méthode proposée a été réalisée en utilisant que des bases de données publiques. La première vidéo de test (AVSS) appartient à l'ensemble de données publiées lors de la Conférence internationale AVSS en 2007¹; la scène représente une station de métro. La deuxième séquence vidéo (CAVIAR) est tirée de la base de données publique CAVIAR², largement utilisée pour l'évaluation des systèmes de surveillance vidéo; la scène montre un centre commercial. Enfin, les deux dernières vidéos sont extraites de la base de données publique PETS2006³, les vidéos montrant le hall d'une gare, à partir de deux angles différents.

Pour chaque vidéo, la vérité terrain a été réalisée par inspection des objets dans chaque trame et en choisissant à la main la ligne de coupe idéale.

Dans le cas idéal, après l'application de l'approche proposée, la valeur résiduelle de la hauteur du reflet $h_r(i)$ doit être égale à zéro. Toutefois, dans le cas réel, nous devons considérer les types d'erreurs suivants :

- l'algorithme ne parvient pas à supprimer complètement le reflet d'un objet détecté; nous appelons ce phénomène *undercut*;
- l'algorithme supprime complètement le reflet, mais aussi une partie de l'objet; nous appelons cette situation *overcut*.

1. <http://www.eecs.qmul.ac.uk/%7Eandrea/avss2007%5Fd.html>

2. <http://homepages.inf.ed.ac.uk/rbf/CAVIARDATA1/>

3. <http://www.cvg.rdg.ac.uk/PETS2006/>

| Séquence | N. Trames | N. Objets | MAE_b | MAE_a | I |
|----------|-----------|-----------|---------|---------|-------|
| AVSS | 5474 | 2531 | 0.160 | 0.053 | 66.9% |
| CAVIAR | 389 | 349 | 0.262 | 0.063 | 75.9% |
| PETS-1 | 3021 | 1610 | 0.601 | 0.286 | 52.4% |
| PETS-2 | 3021 | 2931 | 0.071 | 0.018 | 74.6% |

TABLE 6.2 – Les valeurs du MAE avant et après l’application de la méthode proposée, et l’amélioration obtenue I .

Nous notons avec $e(i)$ l’erreur dans l’estimation de la hauteur de l’ i -ème objet ; il est calculé comme la différence entre la hauteur de l’objet après l’application de la méthode et sa hauteur idéale attendue $h(i)$. Notez que dans le cas de *undercut* $e(i) > 0$, tandis que en présence de *overcut* $e(i) < 0$.

L’indice de performance utilisé est l’erreur absolue moyenne (MAE), qui est la valeur absolue de l’erreur relative $RE(i) = e(i)/h(i)$ en moyenne sur la séquence vidéo :

$$MAE = \frac{1}{N} \cdot \sum_{i=1}^N |RE(i)| \quad (6.6)$$

où N est le nombre d’objets présents dans la séquence.

La Table 6.2 présente les résultats obtenus par la méthode proposée sur les vidéos de test. Pour chaque séquence, nous considérons la valeur du MAE avant (MAE_b) et après (MAE_a) l’application de l’algorithme. Nous présentons également le ratio d’amélioration I , calculé comme suit :

$$I = \frac{MAE_b - MAE_a}{MAE_b} \quad (6.7)$$

Les résultats de la Table 6.2 révèlent une réduction constante de l’erreur due aux reflets sur chaque séquence de test. L’erreur diminue d’environ 70% sur toutes les vidéos, à l’exception de PETS-1 où la baisse d’erreur est légèrement supérieure à 50%. Il est intéressant de noter que la méthode proposée est en mesure de garantir une amélioration significative des performances non seulement pour les vidéos touchées par le problème des reflets, mais aussi sur des vidéos où l’incidence de l’erreur est moins évidente.

Si je devais faire un bilan sur le travail réalisé sur cette problématique je dirai que la méthode est efficace pour la suppression des reflets, mais qu’ils restent certaines contraintes qui la rendent pas suffisamment générale à toutes situations. Par exemple, la méthode ne considère pas les situations dans lesquelles deux personnes sont vues comme une unique région de foreground à cause du reflet : dans ce cas l’algorithme en cherchant la ligne de coupe idéale pour une personne, élimine aussi les pixels appartenant à l’autre personne qui se trouve en dessous de cette ligne. La méthode devrait donc être généralisée à d’autres cas de figure.

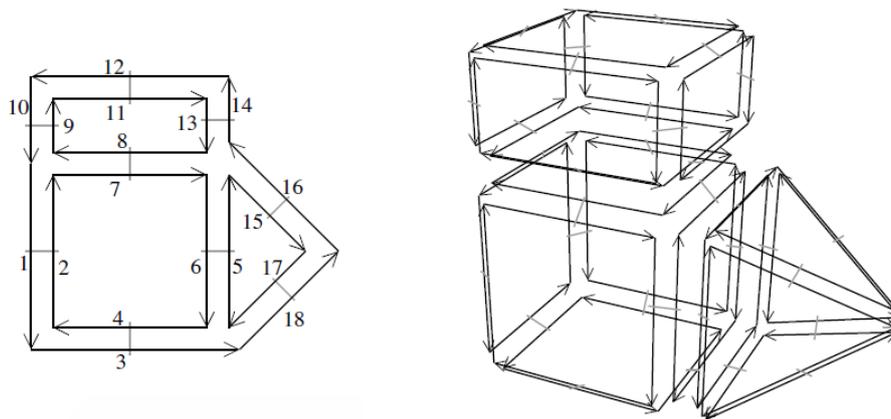


FIGURE 6.7 – Des exemples de cartes combinatoires.

Dé-bruitage des masques de foreground à l'aide des cartes topologiques

Contrairement aux deux autres propositions, cette dernière proposition ne se concentre pas sur un problème spécifique lié à la détection d'objets mais, à travers une représentation structurale, vise à réduire le bruit des masques de foreground indépendamment de la nature du bruit et indépendamment du type de la vidéo. Cette proposition est plus indépendante des caractéristiques de la vidéo. La représentation structurale utilisée dans cette proposition est la carte combinatoire. Cette proposition qui utilise les cartes est le résultat d'une collaboration avec Guillaume Damiand avec lequel j'ai travaillé pendant mon séjour à Lyon en tant que Professeur Invité en 2013; la collaboration a abouti à la publication d'un article [Conte et Damiand (2014)] dont nous présentons ici une synthèse.

Pour comprendre la méthode, il faut d'abord introduire les cartes combinatoires et certaines définitions sur cette structure de données.

Les cartes combinatoires

Une carte combinatoire est définie de la manière suivante (issue de [Lienhardt (1991)]) :

Définition 6.2.1 (Carte combinatoire) Soit $n \geq 0$. Une n -carte combinatoire (ou n -carte) est une algèbre $C = (B, \beta_1, \dots, \beta_n)$ où :

1. B est un ensemble fini de brins ;
2. β_1 est une permutation⁴ sur B ;
3. $\forall i, 2 \leq i \leq n, \beta_i$ est une involution⁵ sur B ;
4. $\forall i, 1 \leq i \leq n - 2, \forall j, i + 2 \leq j \leq n, \beta_i \circ \beta_j$ est une involution.

Les cartes combinatoires permettent de représenter une quasi-variété fermée. Voir la Figure 6.7 pour des exemples de carte combinatoire.

4. Une fonction f de E dans E est une permutation si c'est une bijection de E dans E

5. Une fonction f de E dans E est une involution si c'est une bijection de E dans E telle que $f = f^{-1}$

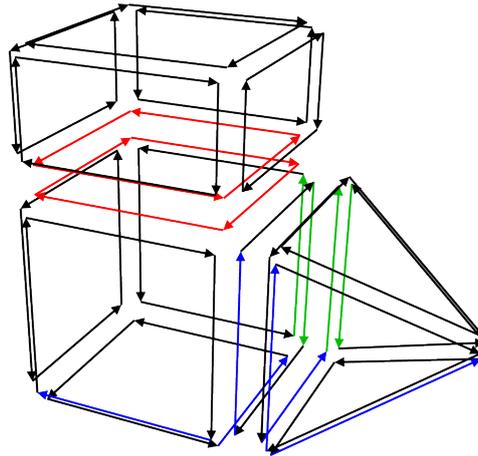


FIGURE 6.8 – Cellules dans une carte combinatoire : en bleu une 0-cellule (sommet), en vert une 1-cellule (arête) et en rouge une 2-cellule (face).

Nous notons β_0 la permutation β_1^{-1} , et β_{ij} la composition $\beta_j \circ \beta_i$. Lorsque deux brins b_1 et b_2 sont tels que $\beta(b_1) = b_2$, nous disons que b_1 est *i-cousu* à b_2 (Damiand (2010b)). Etant donné que les β_i , pour $i \neq 1$, sont des involutions, si b_1 est *i-cousu* à b_2 alors b_2 est *i-cousu* à b_1 . L'opération consistant à mettre en relation deux brins pour β_i est appelée *i-couture*.

Une carte combinatoire est souvent utilisée pour représenter les cellules, c'est-à-dire des complexes simpliciaux. Cependant une carte combinatoire ne représente pas les cellules de manière explicite, mais de manière implicite comme des ensembles de brins obtenus à l'aide de la notion d'orbite. Chaque *i-cellule* est une orbite particulière. Nous notons $\langle \widehat{\beta_{i_1}, \dots, \beta_{i_k}} \rangle$ pour l'orbite contenant toutes les permutations possibles sauf celle notée, c-à-d $\langle \{\beta_1, \dots, \beta_n\} \setminus \{\beta_{i_1}, \dots, \beta_{i_k}\} \rangle$. Nous utilisons également la notation ensembliste $\langle \widehat{I} \rangle$ avec $I = \{\beta_{i_1}, \dots, \beta_{i_k}\}$ pour l'orbite $\langle \widehat{\beta_{i_1}, \dots, \beta_{i_k}} \rangle$. Enfin nous employons indifféremment $\langle \{\beta_{i_1}, \dots, \beta_{i_k}\} \rangle$ ou $\langle \beta_{i_1}, \dots, \beta_{i_k} \rangle$ car cela n'entraîne aucune ambiguïté.

Définition 6.2.2 (*i-cellule*) Soit $C = (B, \beta_1, \dots, \beta_n)$ une n -carte (avec $n > 1$), $b \in B$, et $i \in 0, \dots, n$. La *i-cellule* incidente à b , notée $c_i(b)$, est :

- Si $i = 0$: $\langle \beta_{02}, \dots, \beta_{0n} \rangle$;
- Sinon : $\langle \widehat{\beta_i} \rangle (b)$.

Une *i-cellule* incidente à un brin b peut se voir comme l'ensemble des brins que nous pouvons atteindre par un parcours d'origine b , en utilisant les β donnés dans l'orbite ainsi que leurs inverses. Dit autrement, c'est l'ensemble des brins b' tel qu'il existe un chemin entre b et b' utilisant uniquement les β données ainsi que leurs inverses. Les 0-cellules sont définies ainsi, car nous parcourons uniquement un brin sur deux, afin de n'atteindre que les brins "sortants" du sommet incident à b . Les autres *i-cellules* sont simplement l'orbite composée de tous les β sauf β_i . En effet, comme β_i permet de changer de *i-cellule*, en l'enlevant de l'orbite nous restons au cours du parcours sur les brins de la même *i-cellule*. Voir la Figure 6.8 pour des exemples de cellule sur une carte combinatoire 3D.

Donnons aussi deux autres définitions liées aux cartes généralisées (issues toujours de [Damiand (2010b)]).

Définition 6.2.3 (Incidence) *Deux cellules c_1 et c_2 sont incidentes si elles sont de dimensions différentes, et $c_1 \cap c_2 = \emptyset$.*

Définition 6.2.4 (Adjacence) *Deux cellules c_1 et c_2 sont adjacentes si elles sont de même dimension i , et s'il existe une cellule c de dimension $i - 1$ incidente à c_1 et à c_2 .*

Ce qui est particulièrement intéressant dans les cartes combinatoires est la facilité à calculer des invariants topologiques comme la caractéristique d'Euler-Poincaré ou les nombres de Betti.

La méthode de dé-bruitage

L'idée de base de notre méthode est que le bruit n'est pas facile à détecter et supprimer en analysant la seule image courante de la vidéo (comme le font les autres approches), mais il est plus facile de le détecter si on examine plusieurs frames consécutives. En fait, les objets en mouvement (qui ne sont pas du bruit) au cours de la séquence vidéo présentent une certaine régularité, que le bruit ne semble pas avoir. Par conséquent, notre approche est basée sur une représentation 3D du foreground extrait à partir d'un certain nombre de frames, et la suppression du bruit est faite à l'aide d'opérations d'édition sur cette structure de données. Commençons donc à décrire la représentation 3D proposée.

Dans cette proposition, une séquence temporelle d'images 2D est considérée comme une image 3D. A chaque image de la séquence est associé un instant t . Donc chaque pixel est considéré comme un *pixel temporel*, décrit par trois coordonnées (x, y, t) , (x, y) étant les coordonnées spatiales et t étant la coordonnée temporelle. Dans l'image 3D associée à une séquence d'image 2D chaque voxel représente un pixel temporel.

Une séquence vidéo, représentée par une image 3D décrite ci-dessus, est décomposée en un ensemble de régions 3D qui forment une partition de l'image (c-à-d que chaque voxel appartient à exactement une seule région, et que l'union des régions est égale à l'image entière). La Figure 6.9 montre un exemple de la représentation 3D proposée. La partition forme aussi les cellules (complexes simpliciaux) suivantes : les 0-cellules (sommets), les 1-cellules (arêtes), les 2-cellules (faces) et les 3-cellules (volumes). On retrouve donc les mêmes éléments de la définition de carte combinatoire donnée avant. Nous représentons donc cette subdivision cellulaire avec les cartes topologiques 3D [Damiand (2008)], qui est un modèle 3D, d'une image en 3 dimensions, très efficace, basée sur les cartes combinatoires [Lienhardt (1994), Damiand (2010a)], ou les cellules ainsi que leur relations d'incidence et d'adjacence sont représentées.

Il est important de souligner que cette subdivision cellulaire est une généralisation d'un graphe d'adjacence de régions (RAG [Rosenfeld (1974)]) qui est un graphe ayant un sommet pour chaque région et une arête entre les couples de régions adjacentes. Les sommets du graphes correspondent donc aux régions, alors que les arêtes correspondent aux faces

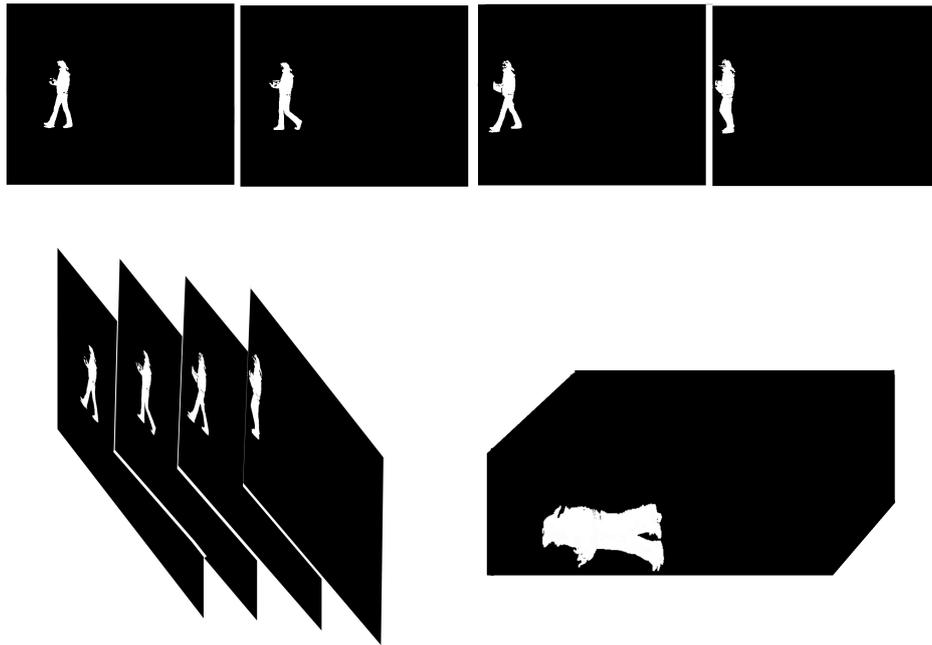


FIGURE 6.9 – Un exemple de représentation 3D d’une séquence vidéo. (a) La séquence d’images. (b) La construction de l’image 3D. (c) La représentation finale.

adjacentes entre deux régions. Comme on peut avoir plusieurs faces adjacentes entre deux régions, cette définition a été étendue aux multi-RAG. Cependant la structure cellulaire est beaucoup plus riche que le RAG ou le multi-RAG, puisqu’elle représente la multi-adjacence et cette relation est ordonnée (étant donnée une région, nous pouvons visiter les régions adjacentes selon un ordre donné, qui n’est pas possible directement avec les graphes); en outre dans notre structure toutes les cellules y sont représentées (alors que dans les RAG seules les 2-cellules et les 3-cellules peuvent l’être).

Avec cette représentation c’est maintenant possible d’utiliser les outils classiques de topologie algébrique, puisque notre subdivision est équivalente à un complexe cellulaire abstrait [Aleksandrov (1961), Kovalevsky (1989)]. Dans notre algorithme on utilisera les nombres de Betti, une invariante topologique connue [Munkres (1984)], pour caractériser la topologie des régions. Les nombres de Betti sont des groupes d’homologies, leur signification intuitive est la suivante : b_0 est le nombre de composantes connexes, b_1 est le nombre de tunnels et b_2 est le nombre de cavités. Pour une région R , $b_0(R) = 1$ puisque par définition une région est connexe, $b_1(R)$ est le nombre de *tunnels* dans R (un tunnel correspond à un chemin de voxels dans R qui ne peut pas être contracté en un point) et $b_2(R)$ est le nombre de *cavités* dans R (une cavité est un ensemble de voxels connexes qui n’appartiennent pas à R mais qui sont complètement entourés de voxels appartenant à R). Une méthode incrémentale pour calculer les nombres de Betti et les mettre à jour lors de la fusion de régions a été développée par Guillaume Damiand et ses collègues [Dupas et Damiand (2009)].

Le principe de notre méthode est détaillé dans l'Algo. 3. Tout d'abord une vidéo est partitionnée en petites séquences consécutives d'images 2D (on ne considère pas les images originales de la vidéo, mais les masques issus de la phase de soustraction de l'arrière plan). Puis chaque séquence est considérée comme une image 3D (comme il a été expliqué avant) et finalement une carte topologique 3D est construite pour représenter la subdivision cellulaire de la séquence considérée. Cette représentation nous donne un ensemble de régions, chacune étiquetée 0 si les voxels correspondent à des voxels de background ou 1 s'ils sont des voxels de foreground.

Algorithme 3 : Algorithme de réduction du bruit dans les masques de foreground

Input : Une vidéo de masques de foreground V ;
 Une fonction booléenne $criterion(r_1, r_2)$.

Result : V est modifiée en fusionnant les couples de régions adjacentes pour lesquelles $criterion$ est satisfaite.

$T \leftarrow$ Construire une carte topologique 3D représentant V ;

foreach région $R_1 \in T$ étiquetée 1, $R_1 \neq R_0$ **do**

foreach région R_2 adjacente à R_1 , $R_2 \neq R_0$ **do**

if $criterion(R_1, R_2)$ **then**

$R \leftarrow$ fusion(R_1, R_2);

mettre à jour la région R ;

return La partition T ;

La carte topologique 3D est créée par l'algorithme proposée dans [Damian (2008)]. Pendant la création, un cube est créé pour chaque voxel, et les voxels 6-adjacents ayant la même étiquette sont fusionnés. En effectuant la fusion pendant la construction, cela nous permet d'élaborer des vidéos de grande taille sans construire le modèle initial composé d'un cube pour chaque voxel, qui rendrait prohibitif, en terme de mémoire, le traitement.

Ensuite, chaque couple de régions adjacentes (R_1, R_2) , avec R_1 étiquetée 1, est considérée. En fait, il suffit de fusionner les régions étiquetées 1 avec le background, sans traiter les régions étiquetées 0. Si un couple (R_1, R_2) vérifie le critère donné, les deux régions sont fusionnées. La fusion est faite par le biais de l'algorithme [Dupas et Damian (2009)] qui est basé sur la suppression des faces communes aux deux régions, en mettant éventuellement à jour les arêtes et les sommets concernés, ainsi que les informations associées à la région fusionnée.

A la fin de l'algorithme, tous les couples de régions ont été considérés et la nouvelle partition de régions est retournée : cette partition décrit la séquence vidéo dans laquelle le bruit a été supprimé.

La complexité de l'Algo. 3 est linéaire avec le nombre d'adjacences entre les régions fois la complexité du critère de fusion.

Maintenant la principale question à aborder est la définition du critère de fusion. En effet ce dernier est l'outil principal pour la réduction du

bruit et seul une définition correcte de ce critère donne des bons résultats. Un premier critère est donné dans l'Eq. 6.8, et il consiste à tester si la taille de la région R_1 de foreground est plus petite qu'un seuil τ défini par l'utilisateur. L'idée à la base de ce critère vient du fait que le bruit dans une séquence d'images produit souvent des petites régions par rapport à la taille des objets réels. Il est important aussi de souligner que pour les objets réels il y a toujours une superposition de l'apparence entre frames consécutives, donc la région 3D représentant l'objet a toujours une taille importante.

$$size(R_1) < \tau \quad (6.8)$$

(R_1 étant la région étiquetée 1)

Le principal intérêt de ce critère est celui d'être très simple à calculer en un temps constant puisque chaque région contient aussi l'information de la taille qui est mis à jour à chaque fusion. Ce première critère n'est cependant pas spécifique aux cartes topologiques, pouvant être calculé aussi sur un multi-RAG. Ici on a pas encore utilisé les informations additionnelles contenues dans les cartes topologiques.

Un problème avec ce premier critère est que certaines régions de foreground peuvent être plus grandes que le seuil, même étant du bruit. En étudiant la nature des régions de bruit, nous avons observé qu'elles sont souvent très poreuses à cause du fait que le bruit n'est pas régulier et les pixels de bruits ne sont pas toujours à la même position dans une séquence vidéo. Par conséquent, ces régions présenteront beaucoup de cavités au contraire des régions décrivant les objets réels qui eux ont généralement un petit nombre de cavités et tunnels. Cette observation nous amène à considérer des régions avec beaucoup de cavités et/ou tunnels comme étant des régions de bruit. On propose donc un deuxième critère de fusion dans l'Eq. 6.9 basé sur la taille et sur les nombres de Betti. Ce critère a deux paramètres : le seuil τ pour les petites régions, et φ , un pourcentage multiplié par la somme des nombres de Betti de R_1 .

$$size(R_1) < \tau * (1 + \varphi * (b_1(R_1) + b_2(R_1))) \quad (6.9)$$

(R_1 étant la région étiquetée 1)

Ce deuxième critère montre l'intérêt d'avoir une représentation plus précise (par rapport aux RAG) des régions, permettant de calculer plusieurs caractéristiques, en particulier des caractéristiques topologiques. La complexité de ce critère est la complexité de calcul des nombres de Betti, c-à-d linéaire avec le nombre de sommets, arêtes et faces de la région R_1 .

Cette approche a été testée sur la base de données publique PETS 2010 [The PETS 2010 dataset (2010)].

Pour évaluer les performances de l'algorithme proposé, nous utilisons en entrée le masque de foreground, résultant de l'application d'un algorithme de soustraction de fond basique. Nous utilisons un algorithme sans aucune optimisation, parce que nous voulons montrer le comportement de l'algorithme de dé-bruitage sans aucun traitement *a priori* de l'image. Cela

nous permet de vérifier si l'algorithme est assez général et s'il n'est pas dépendant de la vidéo. Cela permet aussi d'éviter une étape d'optimisation des paramètres qui est toujours difficile et pas tout le temps possible.

Partant des mêmes masques de foreground, nous comparons l'algorithme avec d'autres propositions que nous avons publiées précédemment et qui s'étaient montrées, par rapport à d'autres approches, plus efficaces (comme on l'a montré dans [Conte et al. (2010a)]). Les algorithmes considérés sont :

- A1** un algorithme de dé-bruitage qui utilise seulement des opérateurs morphologiques d'érosion et dilatation ;
- A2** l'algorithme proposé dans [Conte et al. (2005b)] qui exploite un certain nombre de post-traitements pour la réduction du bruit ;
- A3** l'algorithme proposé dans [Conte et al. (2005b)] auquel on a ajouté l'étape de reconstruction des objets par regroupement de régions voisines décrite dans ce mémoire en cette même Section et publiée dans [Conte et al. (2010a)].

Nous n'avons pas considéré d'autres approches de complexité élevée. Il faut souligner aussi que les algorithmes A2 et A3 requièrent le réglage de plusieurs paramètres. En outre, l'algorithme A3 nécessite, comme on l'a souligné, d'une étape préalable de calibration de la caméra pour chaque vidéo prise avec une caméra différente. Dans nos tests, les paramètres ont été optimisés. Mais il est important de souligner que, surtout l'algorithme A3 dépend fortement des paramètres de la caméra.

La méthode proposée (celle basée sur les cartes combinatoires) dépend de deux paramètres : τ , qu'on a expliqué avant, et nb qui est le nombre de trames considérées pour être représentée par une carte topologique 3D. La méthode basée sur les nombres de Betti a un paramètre additionnel qui est le pourcentage de tunnels/cavités φ .

On utilise un schéma d'évaluation inspiré par la méthode présentée dans [Wolf et Jolion (2004)].

L'objectif du schéma d'évaluation, sur une image, étant donnée la liste des boîtes englobantes des objets de la vérité terrain $G = G_1, \dots, G_n$ et la liste des boîtes englobantes détectées par l'algorithme $D = D_1, \dots, D_m$, est celui de mesurer la qualité des appariements entre les boîtes des deux listes. A partir des deux listes D et G deux matrices σ et τ sont créées. Les lignes $i = 1 \dots |G|$ des matrices correspondent aux objets de la vérité terrain tandis que les colonnes $j = 1 \dots |D|$ représentent les objets détectés.

Les valeurs des matrices sont calculées de la manière suivante :

$$\sigma_{ij} = \frac{\text{area}(G_i \cap D_j)}{\text{area}(G_i)} \quad \tau_{ij} = \frac{\text{area}(G_i \cap D_j)}{\text{area}(D_j)} \quad (6.10)$$

Les matrices sont analysées pour déterminer l'appariement entre les objets des deux listes :

Appariement Un-à-Un : G_i est apparié à D_j si la ligne i des deux matrices contient seulement un élément non-zéro sur la colonne j et la colonne j des deux matrices contient seulement un élément non-zéro sur la ligne i . Il y a donc une seule superposition entre l'objet i de la vérité terrain et l'objet j détecté par l'algorithme ; la surface de superposition nécessite d'avoir une

valeur importante pour que la taille de l'objet détecté soit comparable à celle de l'objet de la vérité terrain ($\sigma_{ij} \geq e_1$ and $\tau_{ij} \geq e_2$).

Appariement Un-à-Plusieurs avec un objet de la Vérité Terrain : G_i est apparié avec plusieurs objets détectés par l'algorithme si la ligne i des matrices contient plusieurs éléments non-zéro. La contrainte additionnelle $\sum_j \sigma_{ij} \geq e_3$ assure que l'objet de la vérité terrain est suffisamment détecté par l'ensemble des objets appariés.

Appariement Un-à-Plusieurs avec un objet détecté par l'algorithme : D_j est apparié avec plusieurs objets de la vérité terrain si la colonne j des matrices contient plusieurs éléments non-zéro. Comme pour les autres cas, nous ajoutons la contrainte $\sum_i \tau_{ij} \geq e_4$.

Les paramètres e_1, \dots, e_4 mesurent la pourcentage de superposition entre les objets de la vérité terrain et ceux détectés par l'algorithme. Pour la plupart des applications une valeur de 0.8 (80% de superposition) est satisfaisante ; pour cette raison nous fixons $e_1 = \dots = e_4 = 0.8$.

En se basant sur cette stratégie d'appariement, les mesures classiques de Rappel et Précision sont définies comme suit :

$$\text{rappel} = \frac{\sum_i \text{Match}_G(G_i)}{|G|} \quad \text{precision} = \frac{\sum_j \text{Match}_D(D_j)}{|D|} \quad (6.11)$$

où $\text{Match}_G(G_i)$ est définie comme suit :

$$\text{Match}_G(G_i) = \begin{cases} 1 & \text{si } G_i \text{ a un appariement un-à-un} \\ 0 & \text{si } G_i \text{ n'a pas d'appariement avec aucun objet } D_j \\ 0.8 & \text{si } G_i \text{ a un appariement un-à-plusieurs} \end{cases} \quad (6.12)$$

et $\text{Match}_D(D_j)$ est définie de manière similaire.

Pour une séquence vidéo on considère la moyenne des valeurs de rappel et précision sur toutes les trames de la séquence.

Les résultats de nos tests sont illustrés dans les Tables 6.3. La Table 6.4 montre l'index F-score. Les cellules en gris foncé représentent la meilleure valeur pour chaque vidéo, et les cellules gris clair la seconde meilleure valeur. Dans ces expériences, nb est toujours fixé à 15. $tXXX$ indique que la méthode a été utilisée avec la valeur du paramètre $\tau = XXX$, et $tXXX-pYYY$ indique que la méthode a été lancée avec les paramètres $\tau = XXX$ and $\varphi = YYY$.

Les résultats montrent que la méthode proposée est compétitive en comparaison des autres approches. Dans la plupart des cas, en fusionnant plus de régions (en augmentant la valeur de τ ou bien de φ) la valeur de Précision augmente tout en diminuant la valeur de Rappel jusqu'à une certaine limite. Donc, le meilleur résultat est obtenu en cherchant le meilleur compromis entre Précision et Rappel.

Les résultats montrent aussi une seconde importante conclusion : la méthode qui utilise les nombres de Betti améliore fortement la performance. C'est le cas par exemple de la vidéo v_7 avec $\tau = 2000$, où la valeur de Précision est augmentée de 0.23 avec le premier critère jusqu'à 0.37 avec le critère utilisant les nombres de Betti ($\varphi = .1$).

Ces résultats sont confirmés par les valeurs de F-score données dans la Table 6.4 qui nous montrent le meilleur compromis entre Precision et Rappel. Le meilleur score est souvent obtenu par la méthode qui utilise les nombres de Betti avec $\tau = 2000$.

| | v1 | | v3 | | v4 | | v5 | |
|------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| | Rec | Pre | Rec | Pre | Rec | Pre | Rec | Pre |
| A1 | 0.55 | 0.09 | 0.38 | 0.27 | 0.54 | 0.24 | 0.55 | 0.16 |
| A2 | 0.44 | 0.29 | 0.20 | 0.39 | 0.46 | 0.36 | 0.44 | 0.32 |
| A3 | 0.49 | 0.22 | 0.31 | 0.42 | 0.52 | 0.33 | 0.50 | 0.20 |
| t2000 | 0.54 | 0.20 | 0.36 | 0.45 | 0.55 | 0.37 | 0.54 | 0.26 |
| t3000 | 0.54 | 0.20 | 0.35 | 0.46 | 0.55 | 0.39 | 0.53 | 0.28 |
| t4000 | 0.53 | 0.21 | 0.33 | 0.46 | 0.54 | 0.40 | 0.53 | 0.28 |
| t2000-p.05 | 0.54 | 0.23 | 0.34 | 0.46 | 0.54 | 0.38 | 0.53 | 0.33 |
| t2000-p.1 | 0.48 | 0.28 | 0.22 | 0.47 | 0.31 | 0.39 | 0.51 | 0.36 |
| t2000-p.15 | 0.39 | 0.36 | 0.08 | 0.34 | 0.09 | 0.37 | 0.45 | 0.35 |
| t3000-p.05 | 0.50 | 0.26 | 0.25 | 0.45 | 0.53 | 0.38 | 0.53 | 0.36 |
| t3000-p.1 | 0.37 | 0.37 | 0.07 | 0.28 | 0.08 | 0.32 | 0.45 | 0.35 |
| t3000-p.15 | 0.23 | 0.37 | 0.03 | 0.14 | 0.03 | 0.17 | 0.32 | 0.30 |
| t4000-p.05 | 0.47 | 0.29 | 0.17 | 0.44 | 0.26 | 0.36 | 0.49 | 0.36 |
| t4000-p.1 | 0.26 | 0.38 | 0.03 | 0.14 | 0.03 | 0.17 | 0.36 | 0.32 |
| t4000-p.15 | 0.14 | 0.40 | 0.01 | 0.07 | 0.01 | 0.06 | 0.21 | 0.22 |

a)

| | v6 | | v7 | | v8 | |
|------------|-------------|-------------|-------------|-------------|-------------|-------------|
| | Rec | Pre | Rec | Pre | Rec | Pre |
| A1 | 0.44 | 0.45 | 0.57 | 0.12 | 0.68 | 0.14 |
| A2 | 0.41 | 0.55 | 0.47 | 0.31 | 0.60 | 0.38 |
| A3 | 0.43 | 0.47 | 0.52 | 0.19 | 0.65 | 0.24 |
| t2000 | 0.46 | 0.50 | 0.55 | 0.23 | 0.66 | 0.25 |
| t3000 | 0.46 | 0.50 | 0.55 | 0.22 | 0.66 | 0.25 |
| t4000 | 0.46 | 0.51 | 0.54 | 0.23 | 0.66 | 0.26 |
| t2000-p.05 | 0.46 | 0.53 | 0.54 | 0.33 | 0.66 | 0.29 |
| t2000-p.1 | 0.44 | 0.54 | 0.46 | 0.37 | 0.56 | 0.32 |
| t2000-p.15 | 0.36 | 0.52 | 0.37 | 0.36 | 0.47 | 0.33 |
| t3000-p.05 | 0.45 | 0.54 | 0.50 | 0.34 | 0.62 | 0.31 |
| t3000-p.1 | 0.35 | 0.52 | 0.36 | 0.37 | 0.46 | 0.33 |
| t3000-p.15 | 0.29 | 0.46 | 0.27 | 0.32 | 0.34 | 0.29 |
| t4000-p.05 | 0.44 | 0.54 | 0.46 | 0.37 | 0.55 | 0.33 |
| t4000-p.1 | 0.32 | 0.48 | 0.29 | 0.34 | 0.39 | 0.30 |
| t4000-p.15 | 0.19 | 0.33 | 0.22 | 0.29 | 0.25 | 0.28 |

b)

TABLE 6.3 – Les valeurs Précision et Rappel pour les algorithmes considérés.

| | v1 | v3 | v4 | v5 | v6 | v7 | v8 |
|-------------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| | <i>Fsc</i> |
| <i>A1</i> | 0.15 | 0.31 | 0.33 | 0.25 | 0.44 | 0.20 | 0.23 |
| <i>A2</i> | 0.35 | 0.26 | 0.40 | 0.37 | 0.47 | 0.37 | 0.46 |
| <i>A3</i> | 0.30 | 0.36 | 0.40 | 0.28 | 0.45 | 0.28 | 0.35 |
| <i>t2000</i> | 0.29 | 0.40 | 0.45 | 0.35 | 0.48 | 0.32 | 0.36 |
| <i>t3000</i> | 0.30 | 0.40 | 0.45 | 0.37 | 0.48 | 0.32 | 0.37 |
| <i>t4000</i> | 0.30 | 0.39 | 0.46 | 0.37 | 0.48 | 0.32 | 0.37 |
| <i>t2000-p.05</i> | 0.33 | 0.39 | 0.45 | 0.41 | 0.49 | 0.41 | 0.40 |
| <i>t2000-p.1</i> | 0.35 | 0.30 | 0.34 | 0.42 | 0.49 | 0.41 | 0.41 |
| <i>t2000-p.15</i> | 0.37 | 0.13 | 0.14 | 0.39 | 0.43 | 0.37 | 0.38 |
| <i>t3000-p.05</i> | 0.34 | 0.32 | 0.44 | 0.43 | 0.49 | 0.41 | 0.41 |
| <i>t3000-p.1</i> | 0.37 | 0.12 | 0.13 | 0.39 | 0.42 | 0.36 | 0.38 |
| <i>t3000-p.15</i> | 0.28 | 0.05 | 0.05 | 0.31 | 0.36 | 0.30 | 0.31 |
| <i>t4000-p.05</i> | 0.36 | 0.25 | 0.30 | 0.41 | 0.49 | 0.41 | 0.41 |
| <i>t4000-p.1</i> | 0.31 | 0.05 | 0.05 | 0.34 | 0.38 | 0.31 | 0.34 |
| <i>t4000-p.15</i> | 0.20 | 0.02 | 0.01 | 0.21 | 0.24 | 0.25 | 0.26 |

TABLE 6.4 – Les valeurs du F-score pour les algorithmes considérés.

6.3 PERSPECTIVES

Dans ce chapitre, nous avons illustré la problématique de la détection d'objets qui est la première étape dans un système de vidéo surveillance.

A partir de la discussion sur l'état de l'art nous avons montré que la méthode la plus utilisée est celle de la soustraction de fond. Ils existent beaucoup de propositions qui essaient d'avoir une représentation riche pour le background et un algorithme de mise à jour efficace et rapide. Malheureusement aucun algorithme ne peut résoudre tous les problèmes liés à la détection d'objets. De plus, les algorithmes les plus efficaces sont aussi les plus complexes.

La direction est alors celle de renoncer à avoir un algorithme idéal de background subtraction mais plutôt d'accepter d'avoir rapidement un masque de foreground issue d'un algorithme basique de détection et d'appliquer des post-traitements pour détecter et supprimer les erreurs présentes. C'est dans cette voie que se situent mes travaux : nous avons présenté des post-traitements pour la suppression des reflets et pour la récupération des objets qui ont des problèmes dus au camouflage. Enfin nous avons présenté une méthode qui réduit le bruit sans aucune information supplémentaire sur le type de bruit ; cette méthode est basée sur les cartes combinatoires.

On peut envisager de nombreuses perspectives liées à la méthode basée sur les cartes. Une première perspective est celle de concevoir une méthode de calcul automatique des paramètres de l'algorithme. Une seconde perspective est celle de définir d'autres critères de fusions des régions : grâce à la représentation par carte topologique, beaucoup de possibilités peuvent être étudiés en mélangeant des critères géométriques et topologiques. Enfin on envisage de concevoir une méthode pour la construction de la carte topologique en ligne, c'est-à-dire au fur et à mesure que les images de la vidéo se présentent à l'application. Pour le moment, la méthode fonctionne hors-ligne sur des vidéos précédemment enregistrées.

Une autre perspective plus générale dans le cadre de la détection d'objets pourrait être la conception d'une méthodologie de mise à jour de l'arrière plan basée sur une sorte de rétro-action. A savoir, étant donnée la solution de détection sur une trame à l'instant t , avec les méthodes de post-traitements présentées nous pouvons imaginer posséder un masque de foreground assez fiable et avec un taux d'erreur réduits. On pourrait donc utiliser ce résultat dans la trame à l'instant $t + 1$ en exploitant la propriété de cohérence temporelle : le masque de foreground d'une trame à l'autre ne peut pas changer beaucoup, donc dans la nouvelle image la solution précédente pourrait être utilisée pour renforcer/ajuster la solution de détection et en même temps pour mettre à jour de manière plus fiable l'arrière-plan. On pourrait imaginer aussi utiliser la solution de détection à l'instant $t + 1$ pour corriger le foreground à l'instant t et puis de ré-utiliser cette solution pour corriger le masque à l'instant $t + 1$ et ainsi de suite jusqu'à un certain degré de fiabilité de la solution, d'une façon similaire aux principes de l'algorithme EM.

SUIVI D'OBJETS

7

SOMMAIRE

| | | |
|-----|---|----|
| 7.1 | ÉTAT DE L'ART | 77 |
| 7.2 | UNE CONTRIBUTION : SUIVI BASÉ SUR L'APPARIEMENT DE PYRAMIDES DE GRAPHES | 79 |
| 7.3 | PERSPECTIVES | 84 |
| 7.4 | CONCLUSIONS | 88 |

LE but de la phase de suivi d'objets dans un système de vidéo surveillance est de calculer les trajectoires des objets en mouvement. En particulier, cette étape traite du problème de la préservation de l'identité d'un objet au cours de la séquence des images, dans lequel l'objet est présent. Le problème du suivi d'objets, de manière générale, peut être défini comme suit : étant donné une image I_{t-1} dans laquelle les objets en mouvement ont une étiquette, le but est de trouver une correspondance entre ces objets et ceux détectés dans l'image courante I_t afin de leur donner l'étiquette (identité) appropriée.

J'ai étudié le suivi d'objets depuis ma thèse et je continue ma recherche dans ce domaine puisque le problème reste encore ouvert aujourd'hui. Le problème du suivi d'objets est peut-être le plus important dans la chaîne de traitement dans un système de vidéosurveillance. Il existe de nombreuses méthodes pour la résolution du problème de suivi comme nous allons le montrer dans l'état de l'art de la Section 7.1. Nous avons proposé en 2006 [Conte et al. (2006)] un algorithme basé sur l'appariement de pyramides de graphe (présenté dans le chapitre 3) qui reste encore efficace aujourd'hui en terme de qualité du suivi. Cette méthode sera présentée dans la Section 7.2.

Parmi les perspectives qui concernent le suivi d'objets (Section 7.3) nous allons en présenter une en particulier déjà bien avancée puisque seule la phase d'expérimentation reste manquante. Il s'agit d'une nouvelle méthode de suivi basée sur les filtres particuliers et sur les graphes, qui

exploite les avantages des méthodes probabilistes, dans un cadre structurale, pour améliorer les performances de l'état de l'art.

7.1 ÉTAT DE L'ART

Les travaux qui abordent le suivi d'objets se différencient par la façon de mettre en correspondance les objets de deux trames successives. Beaucoup de travaux effectuent le suivi après la phase de détection des objets et donc la correspondance est faite à l'aide du masque de foreground calculé à l'étape précédente. Haritaoglu et al. (2000) proposent une simple, mais efficace, mesure de correspondance : la superposition entre les boîtes englobantes des objets des deux trames. Ainsi, un objet de l'image courante obtient l'étiquette de l'objet de l'image précédente, pour lequel la superposition entre les surfaces est la plus grande. Cette méthode marche bien quand le nombre d'images par seconde de la vidéo est suffisamment élevé. D'autres méthodes ([Chen et al. (2012), Jiang et al. (2011)]) utilisent aussi l'information de la position des objets et la correspondance est faite à l'aide de la prédiction de la position dans l'image successive à travers les filtres de Kalman. Cette approche n'est pas très fiable en présence de mouvements non-linéaires. Dai et al. (2007) ont proposé une méthode de mesure de correspondance basée sur la forme des objets. Pour faire face à la correspondance de plusieurs objets à suivre et dans un souci d'avoir un matching plus global, Zhang et al. (2012) proposent d'utiliser la méthode hongroise sur une représentation par graphe bipartite des objets présents dans les deux images. La proposition de Pellegrini et al. (2009) essaie de prévoir les correspondances à l'aide d'un ensemble de modèles de comportement appris en utilisant des vidéos différentes de celles du test. La méthode est efficace mais nécessite une phase d'apprentissage et peut avoir des problèmes pour des configurations d'objets pas vues dans la phase d'apprentissage. D'autres méthodes ([Lin et Huang (2011), Berclaz et al. (2011)]) se basent sur les informations prises par plusieurs caméras pour résoudre le problème du suivi. Mais ces techniques ne sont pas utilisables dans des situations où une seule caméra est disponible.

Finalement, le problème principal dans le suivi d'objets est sûrement le problème des occlusions. Une occlusion se produit lorsqu'un objet est couvert soit par un élément statique de la scène (par exemple un arbre ou un mur) ou par un autre objet en mouvement. A cause de la représentation 2D de l'image, la phase de détection d'objets repère une unique région de foreground dans laquelle plusieurs objets en font partie. Dans ce cas, la correspondance un-à-un entre les régions de foreground de deux images consécutives n'est pas une solution correcte.

Une approche différente de celle décrite précédemment est celle qui ne se base pas sur une étape de détection ; le suivi est fait sur la base de la recherche d'un modèle d'objet à suivre, directement sur l'image courante de la vidéo ; souvent le modèle est mis à jour le long de la séquence pour s'adapter aux changements. L'article de Comaniciu et al. (2000) propose l'utilisation de l'algorithme Mean Shift, un algorithme itératif rapide pour trouver le barycentre d'une distribution de probabilité, afin de déterminer la position la plus probable de l'objet à suivre. Cet objet est sé-

lectionné manuellement. La méthode gère les occlusions partielles, mais pas les occlusions totales (quand l'objet est complètement caché par un autre). Tao et al. (2002) ont proposé une méthode basée sur une représentation en couches de la scène, chaque couche représentant un objet en mouvement. L'apparence des couches sont créées et mises à jour en utilisant un cadre probabiliste. Leur méthode est en mesure de traiter les occlusions, mais elle est très coûteuse. La méthode proposée par Bhuvaneshwari et Rauf (2009) utilise des caractéristiques basées sur les contours appelées "edglet" et un ensemble de classifieurs pour détecter les occlusions partielles ; le suivi est basé sur l'utilisation d'un filtre de Kalman. La méthode ne gère pas les occlusions totales et, en raison du filtre de Kalman, elle fonctionne surtout pour des mouvement de type linéaire (vitesse et direction constante). Les méthodes plus prometteuses dans cette catégorie d'approches sont sûrement celles basées sur les filtres particuliers. Les filtres particuliers (ou filtres à particules) se basent sur une représentation approximée d'une distribution de probabilité en utilisant un ensemble fini d'échantillons. Les filtres à particules semblent très prometteurs car ils peuvent traiter les mouvements non-linéaires. Les travaux les plus significatifs dans cette catégorie sont les suivants : la première proposition d'application des filtres particuliers au suivi d'objets a été celle de Isard et Blake (1998) qui a servi de base pour les propositions suivantes. Différentes adaptations du filtre pour faire face au suivi multi-objets ont suivi : dans Doucet et al. (2002), les auteurs ont proposé un système appelé Jump Markov System (JMS) qui modélise et estime conjointement les états de l'objet, le nombre d'objets à suivre et l'association entre estimation et mesure des objets. Dans Okuma et al. (2004), les auteurs modélisent la distribution de probabilité par un mélange de distributions afin de gérer plusieurs objets, et ils utilisent la procédure de Adaboost [Freund et Schapire (1997)] pour détecter les objets qui entrent et sortent des images. Afin de mieux modéliser les interactions entre les objets, les auteurs de [Widynski et al. (2012)] ont proposé l'intégration de contraintes spatiales floues dans le cadre des filtres particuliers pour le suivi multi-objets.

Même dans la deuxième catégorie d'approches (suivi sans la détection) les occlusions restent un problème dans la phase d'association entre le modèle à suivre et l'observation courante. En effet, souvent, ces techniques n'arrivent pas à gérer les occlusions totales. Les filtres particuliers dans ce cas peuvent être efficaces grâce à une bonne estimation des états cachés qui peuvent représenter aussi des configurations qui ne sont pas observables sur l'image. Cependant, les filtres particuliers dans le suivi multi-objets souffrent d'autres problèmes.

Un problème majeur est celui de la malédiction de la dimension [MacKay (1998)]. Cela signifie que le filtre à particules nécessite un certain nombre de particules qui augmente de façon exponentielle avec le nombre d'objets, ce qui rend l'utilisation de ce type de filtres impossible dès que le nombre d'objets est supérieur à trois.

Le principe de l'échantillonnage partitionné est de partitionner l'espace d'état, en prenant en compte un élément de la partition par objet. Les objets sont traités dans un ordre préétabli et les particules les plus

susceptibles de correspondre à l'état réel de l'objet sont sélectionnées selon une méthode d'échantillonnage pondérée. La question de l'ordre peut conduire à des comportements inappropriés du filtre, par exemple lorsque le premier objet est caché. Traiter plusieurs objets dans un filtre à particules soulève un autre type de problème. Il est souvent nécessaire de modéliser les interactions possibles entre les objets, afin d'estimer conjointement les états.

Dans la section des perspectives nous allons décrire une proposition qui se base sur les filtres particulières, puisqu'il semble s'agir de l'approche récente la plus prometteuse, et qui essaye de remédier aux problèmes décrits.

Avant cela, dans la prochaine section, nous allons décrire une proposition déjà publiée ([Conte et al. (2006)]) qui se situe dans la première catégorie d'approches : cette méthode nécessite une phase de détection et essaye de trouver une correspondance entre les objets de deux images consécutives. Comme nous l'avons expliqué précédemment, la limite de la première approche réside dans la nécessité de choisir la bonne granularité de la décomposition en régions. Avec une décomposition en régions trop fine, l'avantage de simplicité des calculs de cette approche est perdu. Inversement si on a une partition des régions à gros grain, le système n'est pas suffisamment robuste pour faire face aux occlusions.

Nous avons proposé un algorithme basé sur une représentation des objets par pyramide de graphes, en utilisant une approche multi-résolution qui sait exploiter la cohérence spatiale entre les pixels. La représentation basée sur les graphes permet une description des objets et de leurs relations mutuelles assez riche. Le principal avantage de la méthode proposée est qu'elle utilise dans chaque cas, le minimum de détails nécessaires pour effectuer sa tâche : dans les cas simples, quand il n'y a aucune ambiguïté (pas d'occlusions), seule une description grossière de la scène (correspondant au niveau le plus haut des pyramides de graphes) est utilisée, fournissant ainsi une solution très rapide. Par contre, pendant les occlusions, l'algorithme est capable d'exploiter des détails plus fins et complexes (représentés par les niveaux inférieurs des pyramides de graphes) pour assigner les pixels de l'image à chaque objet participant à l'occlusion.

Il est aussi important de noter que ce procédé de raffinement de la granularité n'est pas nécessairement réalisé sur l'objet dans son ensemble, mais seulement sur les parties de l'objet qui, au niveau de résolution courante, présentent des ambiguïtés en ce qui concerne le suivi.

7.2 UNE CONTRIBUTION : SUIVI BASÉ SUR L'APPARIEMENT DE PYRAMIDES DE GRAPHES

Chaque région du premier plan est représentée par une pyramide irrégulière de graphes (décrite dans le chapitre 3). Chaque niveau de la pyramide représente une décomposition d'une région en sous-régions correspondant à une granularité différente. À savoir, le plus haut niveau L_1 est un graphe d'un seul sommet représentant la région dans son ensemble. Dans les niveaux suivants, chaque sommet représente une sous-région, tandis que les arêtes représentent la relation d'adjacence entre les sous-

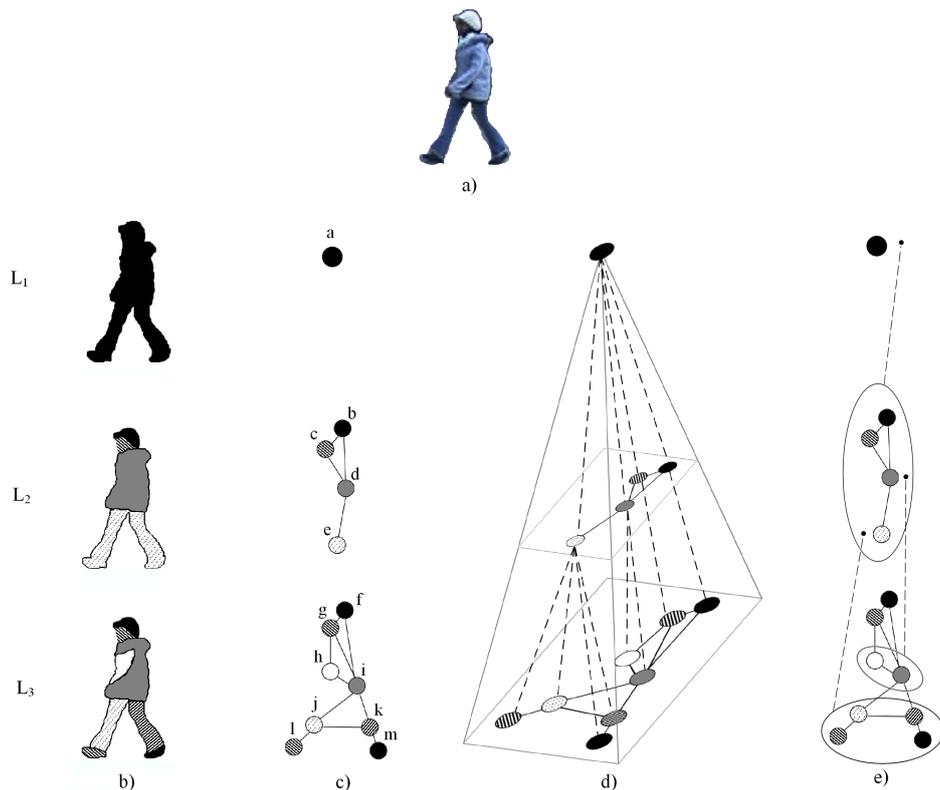


FIGURE 7.1 – La représentation d'une région de premier plan par une pyramide de graphe. (a) La région représentée. (b) La segmentation de la région à différents niveaux de résolution. (c) Les graphes correspondants pour chaque niveau. Les attributs associés à cet exemple sont indiqués dans la Table 7.1. (d) La structure hiérarchique de la pyramide. (e) Une version graphique simplifiée de la pyramide.

régions. On suppose que le parent d'un ensemble de sommets à un niveau donné de la pyramide représente une région qui est l'union des sous-régions représentées par tous ces sommets; à l'inverse, les descendants d'un sommet représentent une partition de la région représentée par leur ascendant.

Chaque sommet n dans la pyramide contient une description de la région (ou sous-région) associée par le biais des attributs suivants :

- les coordonnées du centre de la boîte englobante de la région $c_x(n)$ et $c_y(n)$;
- la taille de la boîte englobante $w(n)$ et $h(n)$;
- la moyenne de couleur des pixels de la sous-région μ_R, μ_G, μ_B .

Un exemple de pyramide de graphe, avec les attributs sur les sommets, est montré en Figure 7.1 et Table 7.1. La pyramide est construite par une méthodologie bottom-up, en utilisant l'algorithme de décimation-regroupement décrit dans [Jolion et Montanvert (1992)].

| Node ID | c_x | c_y | w | h | μ_R | μ_G | μ_B |
|---------|-------|-------|-----|-----|---------|---------|---------|
| a | 219 | 267 | 217 | 374 | 190 | 196 | 210 |
| b | 223 | 123 | 72 | 54 | 180 | 191 | 208 |
| c | 201 | 147 | 54 | 42 | 56 | 58 | 68 |
| d | 223 | 235 | 118 | 162 | 99 | 121 | 163 |
| e | 212 | 393 | 217 | 160 | 60 | 72 | 105 |
| f | 223 | 123 | 72 | 54 | 180 | 191 | 208 |
| g | 201 | 147 | 54 | 42 | 56 | 58 | 68 |
| h | 205 | 227 | 60 | 86 | 75 | 95 | 138 |
| i | 223 | 235 | 118 | 162 | 105 | 127 | 171 |
| j | 164 | 380 | 91 | 148 | 57 | 70 | 106 |
| k | 271 | 383 | 91 | 136 | 55 | 68 | 103 |
| l | 126 | 460 | 56 | 31 | 46 | 54 | 78 |
| m | 294 | 455 | 62 | 43 | 56 | 70 | 97 |

TABLE 7.1 – Les valeurs des attributs pour les graphes de la Figure 7.1.

Avec cette représentation, le suivi est effectué à l'aide de l'algorithme d'appariement de pyramides décrit dans le chapitre sur l'appariement de graphes (voir Section 3.3.2). Ici, on le particularise pour le cas du suivi d'objets.

Le but de l'algorithme de suivi est de fournir un étiquetage des régions détectées qui soit compatible avec les identités des objets : les régions correspondantes aux mêmes objets dans différentes images de la séquence vidéo doivent obtenir la même étiquette, même si leur position et leur apparence a changé. Inversement, les régions correspondant à des objets différents doivent avoir une étiquette différente, même si elles ont le même aspect.

En cas d'occlusions partielles, une région détectée peut correspondre à plusieurs objets ; afin de faire face à ces situations, notre algorithme attribue une étiquette pour toutes les sous-régions (dans la hiérarchie représentée par la pyramide correspondante). Il peut arriver, en particulier aux niveaux supérieurs de la pyramide, qu'une région contienne des parties appartenant à différents objets ; dans ce cas, il n'est pas possible d'attribuer une étiquette unique à la région. Notre algorithme gère une telle situation en marquant le sommet correspondant 'multilabel'. Ensuite l'algorithme, en descendant à des niveaux de granularité plus fins, est capable d'arriver à trouver une correspondance de tous les sommets d'une pyramide avec les sommets correspondants dans une autre pyramide de manière à étiqueter les sous-régions d'une région 'multilabel' avec une étiquette unique (un seul objet par sous-région).

La correspondance entre régions dans les différents niveaux est effectuée à l'aide d'un algorithme d'appariement de graphes, à savoir la recherche du plus grand sous-graphe commun à deux graphes.

Les Figures 7.2 et 7.3 montrent, respectivement, les pyramides corres-

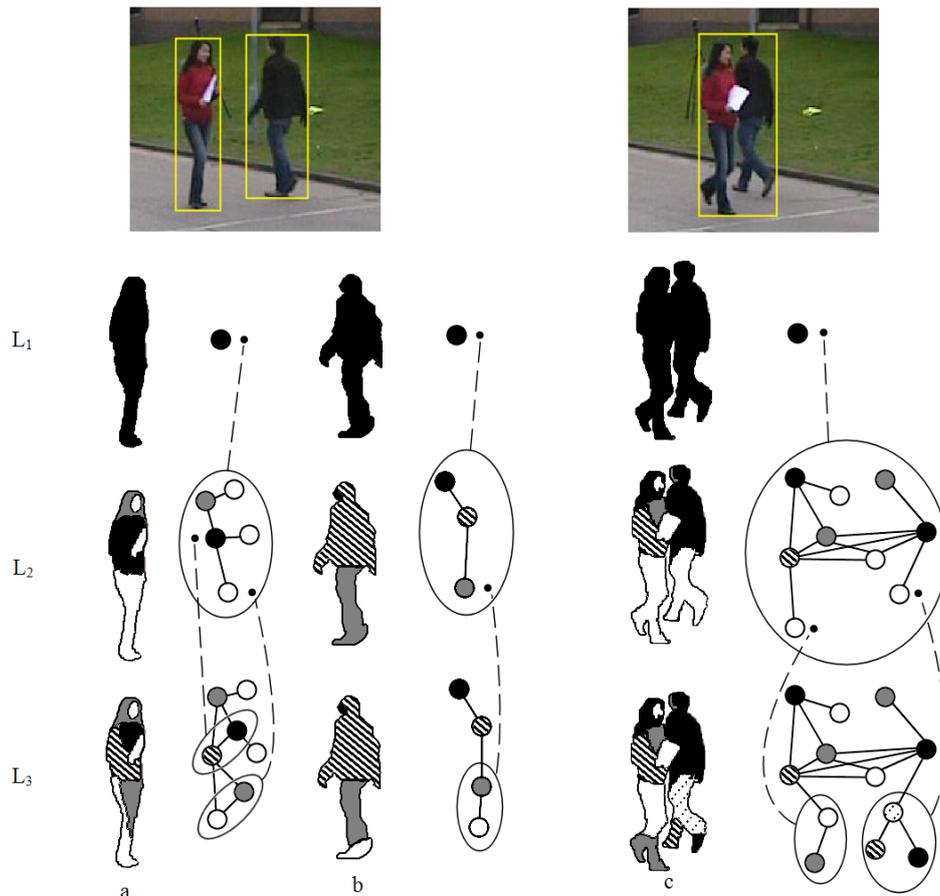


FIGURE 7.2 – Les frames considérés et la pyramide de graphes correspondant. Dans le frame de gauche, deux régions a et b ont été détectées. Chaque région est comparée à la région c du frame de droite.

pendant aux régions de deux frames consécutifs et le fonctionnement de l'algorithme de suivi.

Les expérimentations ont été effectuées sur la base de données de la conférence PETS déjà citée en considérant l'ensemble de l'édition 2010 [The PETS 2010 dataset (2010)]. Cette dernière version a été expressément pensée pour tester les algorithmes de suivi (les années suivantes les organisateurs de la conférence PETS ont créé d'autres benchmark pour d'autres problématiques d'analyse de vidéo). Par souci de comparabilité avec les autres méthodes, nous avons re-testé l'algorithme en utilisant les indices de performances conçus spécifiquement par la communauté des chercheurs dans le suivi d'objets [Kasturi et al. (2009)]. Ces indices sont désormais une référence pour la comparaison des algorithmes et ils n'avaient pas encore été publiés en 2006.

Les résultats de cette expérience sont résumés dans la Table 7.2. A noter que pour tous les indices considérés, une valeur élevée correspond à une meilleure performance. La méthode proposée montre des performances comparables à d'autres méthodes et dans de nombreux cas l'algorithme surpasse les autres approches (qui sont considérées à ce jour comme les meilleures dans le domaine).

Certaines astuces ont été ajoutées (comme le fait de ne pas reconstruire les pyramides des objets à chaque frame) permettent d'exécuter l'algo-

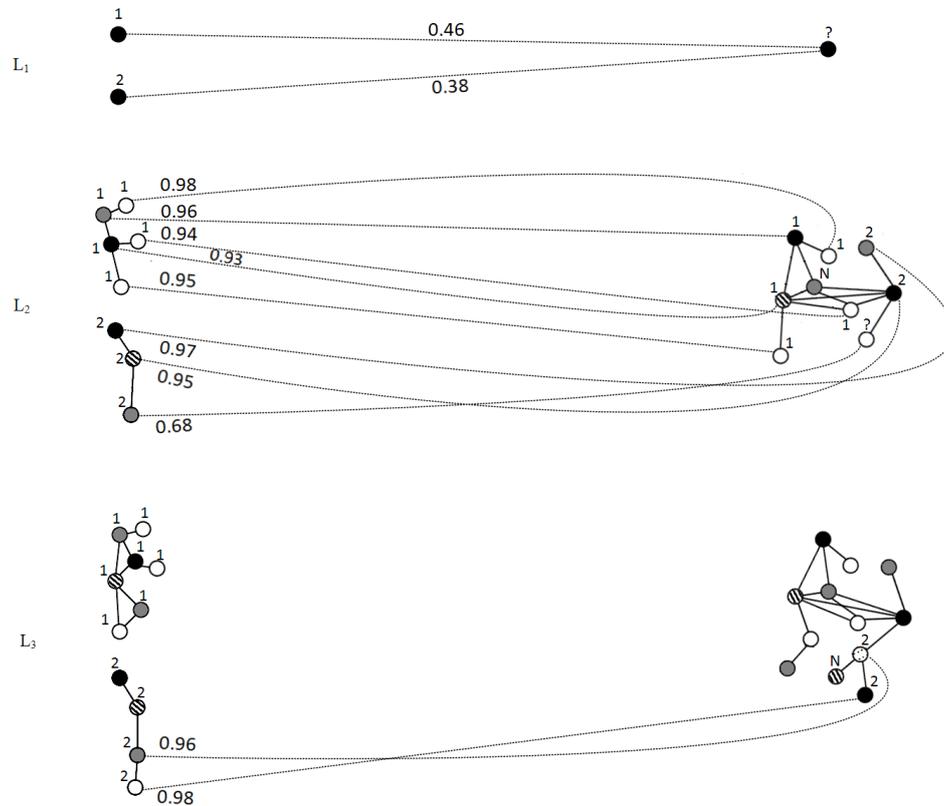


FIGURE 7.3 – L'application de l'algorithme d'appariement sur les pyramides de la Figure 7.2. Les lignes en pointillés représentent le résultat de l'appariement de graphe.

Au niveau L_1 , aucune des régions a et b est suffisamment similaire à c et donc l'algorithme continue au niveau suivant. Au niveau L_2 , la plupart des sous-régions ont été étiquetées grâce aux valeurs de similarité ; une sous-région est marquée 'N' parce que aucun appariement n'a été trouvé ; cette région sera étiquetée de la même manière que le sommet adjacent le plus similaire. Au niveau L_3 l'étiquetage des sous-régions est complété. Ainsi, les étiquettes des niveaux inférieurs peuvent se propager vers le haut.

TABLE 7.2 – Résultats expérimentaux. La table montre la valeur médiane des indexes sur toutes les séquences vidéo de la base de données PETS2010 pour les algorithmes considérés.

| Algorithme | ATA | MOTA | MOTP | MODA | MODP | SFDA |
|-------------------------|-------------|-------------|-------------|-------------|-------------|-------------|
| Berclaz et al. (2009) | 0.14 | 0.71 | 0.55 | 0.72 | 0.55 | 0.48 |
| Arsic et al. (2009) | 0.01 | 0.21 | 0.39 | 0.24 | 0.39 | 0.18 |
| Conte et al. (2010e) | 0.03 | 0.55 | 0.58 | 0.60 | 0.58 | 0.46 |
| Di Lascio et al. (2012) | 0.07 | 0.61 | 0.60 | 0.60 | 0.62 | 0.50 |
| Méthode proposée | 0.09 | 0.6 | 0.71 | 0.59 | 0.71 | 0.57 |

rithme avec un coût de calcul raisonnable : l'application de l'algorithme est possible sur un taux d'images par second de 1 ou 2 FPS. Néanmoins, 8 ans après la publication de la méthode, on pourrait enquêter sur d'autres améliorations pour la réduction des temps d'exécution.

7.3 PERSPECTIVES

A partir de la considération faite à la fin de la section précédente une première perspective est d'étudier d'autres représentations hiérarchiques des objets (par exemple les cartes topologiques) en exploitant la même idée de l'algorithme d'appariement proposé mais sur des structures de données plus riches qui pourraient mieux s'adapter au problème.

Les dernières propositions de la littérature montrent que les techniques des filtres particulières sont les plus prometteuses. A partir de ce constat, j'envisage de proposer des nouvelles méthodes intégrant les graphes dans le cadre des filtres particulières pour faire face aux problèmes ouverts des filtres, illustrés dans la Section 7.1. Cette idée ne constitue pas juste une perspective, mais fait partie des travaux que je mène en ce moment grâce à la collaboration avec Isabelle Bloch de Telecom ParisTech et Séverine Dubuisson de l'Université Paris 6. Les lignes directrices de ces travaux sont fournies dans les sections suivantes.

Application des filtres particulières basés graphes au suivi d'objets

Dans le problème de filtrage classique un système est décrit comme suit [Bishop (2006)] :

$$\mathbf{x}_t = f_t(\mathbf{x}_{t-1}, \mathbf{v}_t) \quad (7.1)$$

$$\mathbf{y}_t = h_t(\mathbf{x}_t, \mathbf{w}_t) \quad (7.2)$$

où \mathbf{x}_t représente l'état caché à l'instant t du système, \mathbf{y}_t la mesure extraite de l'image courante, f_t modèle l'évolution temporelle de \mathbf{x}_t , h_t est l'équation de la mesure, et \mathbf{v}_t et \mathbf{w}_t sont des bruits blancs indépendants.

Le suivi bayésien non-linéaire consiste à estimer la fonction de densité *a posteriori* $p(\mathbf{x}_t | \mathbf{y}_{1:t})$ (où $\mathbf{y}_{1:t}$ indique l'ensemble des mesures de l'instant 1 à l'instant t), exprimée par :

$$p(\mathbf{x}_t | \mathbf{y}_{1:t}) = \frac{p(\mathbf{y}_t | \mathbf{x}_t) \cdot p(\mathbf{x}_t | \mathbf{y}_{1:t-1})}{\int_{\mathcal{X}} p(\mathbf{y}_t | \mathbf{x}'_t) \cdot p(\mathbf{x}'_t | \mathbf{y}_{1:t-1}) d\mathbf{x}'_t} \quad (7.3)$$

où \mathcal{X} est l'espace des états. La densité *a priori* est définie par :

$$p(\mathbf{x}_t | \mathbf{y}_{1:t-1}) = \int_{\mathcal{X}} p(\mathbf{x}_t | \mathbf{x}_{t-1}) \cdot p(\mathbf{x}_{t-1} | \mathbf{y}_{1:t-1}) d\mathbf{x}_{t-1} \quad (7.4)$$

Les filtres particuliers sont utilisés pour approximer la fonction de densité *a posteriori* par une somme pesée de N masses de Dirac $\delta_{\mathbf{x}_t^{(n)}}(d\mathbf{x}_t)$ centrées sur des réalisations hypothétiques $\{\mathbf{x}_t^{(n)}\}_{n=1}^N$ de l'état \mathbf{x}_t , appelés particules [Doucet et al. (2001)].

Donc, la distribution $\mathbb{P}(\mathbf{x}_t | \mathbf{y}_{1:t})$ est récursivement approximée par la distribution empirique $P_N(\mathbf{x}_t | \mathbf{y}_{1:t}) = \sum_{n=1}^N w_t^{(n)} \delta_{\mathbf{x}_t^{(n)}}(\mathbf{x}_t)$, où $\mathbf{x}_t^{(n)}$ est la $n^{\text{ème}}$ particule et $w_t^{(n)}$ son poids.

Si une approximation de la fonction $\mathbb{P}(\mathbf{x}_t | \mathbf{y}_{1:t})$ est connue, le processus est divisé en trois étapes principales :

1. L'étape de diffusion consiste à estimer $p(\mathbf{x}_t | \mathbf{y}_{1:t-1})$ en propageant la nuée de particules $\{\mathbf{x}_t^{(n)}, w_t^{(n)}\}_{n=1}^N$ en utilisant une fonction d'importance $q(\mathbf{x}_t | \mathbf{x}_{0:t}^{(n)}, \mathbf{y}_t)$.
2. L'étape de mise à jour calcule ensuite le poids des nouvelles particules à l'aide des nouvelles observations \mathbf{y}_t , c'est-à-dire :

$$w_t^{(n)} \propto w_{t-1}^{(n)} \frac{p(\mathbf{y}_t | \mathbf{x}_t^{(n)}) p(\mathbf{x}_t^{(n)} | \mathbf{x}_{t-1}^{(n)})}{q(\mathbf{x}_t | \mathbf{x}_{0:t-1}^{(n)}, \mathbf{y}_t)} \quad (7.5)$$

tel que $\sum_{i=1}^N w_t^{(i)} = 1$

3. Une étape de ré-échantillonnage est utilisée pour éviter les problèmes de la dégénérescence des particules [Doucet et al. (2001)].

Reprenons donc les problèmes des filtres particulaires appliqués au suivi multi-objet. Premièrement, le problème d'association des mesures aux objets est généralement complexe et mène à des algorithmes qui ne le sont pas moins, ce qui augmente considérablement la complexité algorithmique des filtres. En outre modéliser les interactions entre objets soulève la question de la formulation d'hypothèses que l'on effectue sur la scène, ce qui reste encore à formaliser. Le dernier problème est celui de la dimension de l'espace d'état. Cet enjeu concerne tous les filtres particulaires dont l'état est multi-dimensionnel, et donc particulièrement ceux multi-objets, dans lequel les états des objets sont concaténés pour finalement former le vecteur aléatoire inconnu du système.

Avec la proposition suivante nous voulons essayer de résoudre ces problèmes.

Nous proposons un filtre particulaire dans lequel les états et les observations sont représentées par des graphes. Par conséquent, les objets à suivre ne sont pas considérés d'une manière séquentielle, mais un seul état représente toute la scène, où l'évolution des objets se produit simultanément à l'intérieur de la représentation par graphe. Ainsi nous pensons résoudre le problème d'association des mesures aux objets (on n'a pas différentes mesures à associer aux différents objets, mais une seule observation représentée par un graphe). De même, le problème de la modélisation des interactions entre objets est résolu par la structure du graphe et le problème dimensionnel est sûrement réduit parce que en ayant un seul état pour tous les objets la dimension de l'espace est plus petit que dans le cas où on utilise un état pour chaque objet à suivre.

Nous allons décrire la représentation de l'état-caché et de l'observation par un graphe. Étant donnée une image à l'instant t , nous représentons les objets présents dans l'image par un graphe comme suit : chaque sommet

représente un objet et chaque arête représente la relation spatiale entre deux objets.

Pour construire le graphe à partir d'une image, une étape préliminaire de détection d'objet est exécutée (voir le Chapitre 6).

Pour les sommets et les arêtes, on utilise les attributs suivants :

- chaque sommet représente un objet avec les attributs suivants :
 - l'ID de l'objet ;
 - la visibilité de l'objet (un boolean qui indique si l'objet est visible ou caché par un autre objet dans le cas d'occlusion) ;
 - la taille (w, h) de la boîte englobante de l'objet ;
 - les coordonnées (c_x, c_y) du centre de la boîte englobante de l'objet ;
 - la dynamique de la taille (\dot{w}, \dot{h}) ;
 - la dynamique de la position (\dot{c}_x, \dot{c}_y) ;
 - l'histogramme des trois canaux de couleur $h_o(A)$ de l'image associé à l'objet (A représente l'ensemble des pixels de la boîte englobante de l'objet)
- une arête représente la relation spatiale entre deux objets ; soient R_1 et R_2 les deux régions correspondantes aux objets dont on veut calculer la relation spatiale ; l'attribut des arêtes est une mesure basée sur la notion d'adjacence floue de la manière suivante :
 - Une mesure de distance peut être définie sur un intervalle flou f de forme trapézoïdale sur \mathbb{R}^+ ;
 - Un sous-ensemble flou μ_d sur l'espace image \mathcal{S} peut être ensuite dérivé en combinant f avec une carte de distance d_A par rapport à un objet de référence $A : \forall X \in \mathcal{S}, \mu_d(x) = f(d_A(x))$, où $d_A(x) = \inf_{y \in A} d(x, y)$;
 - En considérant une des deux régions comme référence, disons R_1 , la mesure d'adjacence floue entre R_1 et R_2 est (voir [Bouchon-Meunier et al. (1996)]) :

$$adj(near(R_1), R_2) = \frac{\sum_{x \in R_2} \mu_{near(R_1)}(x)}{\sum_{x \in \mathcal{S}} \mu_{R_2}(x)}$$

où $\mu_{R_2}(x)$ est le degré d'appartenance du pixel x à la région R_2 . Cette mesure représente la partie de l'objet R_2 contenue dans

$\mu_{near(R_1)}$.

- une arête est présente entre deux sommets uniquement lorsque la mesure d'adjacence va au-delà d'une valeur minimale τ . On considère que, au-dessous d'un certain seuil, il n'y a aucune relation spatiale entre les deux régions.

Il est important de souligner que la mesure d'adjacence adoptée, n'est qu'une possibilité parmi d'autres relations spatiales qui peuvent également être considérées. La méthode est, en effet, indépendante des attributs choisis pour les sommets et pour les arêtes.

Cette représentation est utilisée pour décrire à la fois l'état caché et l'observation du système. Dans la Figure 7.4 et dans la Table 7.3 un exemple du graphe associé à une image est présenté.

Puisque l'état est représenté par un graphe (appelé *graphe état* ci-après), son évolution est décrite en termes d'opérations sur les graphes. Compte

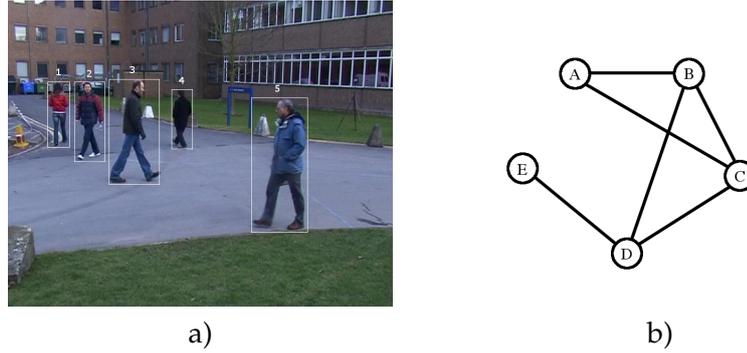


FIGURE 7.4 – a) Une image à l’instant t ; b) Sa représentation par la biais d’un graphe (les valeurs des attributs sont dans la Table 7.3).

tenu du graphe état \mathbf{x}_{t-1} à l’instant $t - 1$, des opérations de mise à jour des attributs et d’ajout/suppression de sommets sont définies selon un certain modèle dynamique.

La fonction de vraisemblance $p(\mathbf{y}_t|\mathbf{x}_t)$ est définie par une mesure de distance entre le graphe état prédit \mathbf{x}_t et le graphe observation \mathbf{y}_t . C’est ici que nous exploitons la technique du plongement de graphe : nous utilisons la distance induite par un noyau k entre les deux graphes, défini comme $d^2(x, y) = k(x, x) + k(y, y) - 2 \cdot k(x, y)$. Avec cette définition, la fonction de vraisemblance devient $p(\mathbf{y}_t|\mathbf{x}_t) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{d(\mathbf{y}_t, \mathbf{x}_t)}{2\sigma^2}}$.

Avec cette fonction de vraisemblance définie, la phase de mise à jour peut-être exprimée en pseudo-code comme suit :

– **for** $n = 1, \dots, N$

$$\tilde{w}_t^{(n)} = w_{t-1}^{(n)} p(\mathbf{y}_t|\mathbf{x}_t)$$

– **for** $n = 1, \dots, N$

$$w_t^{(n)} = \frac{\tilde{w}_t^{(n)}}{\sum_{m=1}^N \tilde{w}_t^{(m)}}$$

Enfin, une technique classique de ré-échantillonnage est utilisée pour éviter les problèmes de dégénérescence des particules. Nous avons choisi de ré-échantillonner les particules selon une distribution multinomiale avec les paramètres N et $\{w_t^{(n)}\}_{n=1}^N$.

Comme estimateur du filtre (la sortie de l’algorithme), nous choisissons le maximum de la distribution a posteriori approximée par les particules (avant la phase de ré-échantillonnage) :

$$\hat{\mathbf{x}}_t^{MAP} = \operatorname{argmax}_{\mathbf{x}_t} \sum_{n=1}^N w_t^{(n)} \delta_{\mathbf{x}_t^{(N)}}(d\mathbf{x}_t) \quad (7.6)$$

| <i>Attributs des sommets</i> | |
|------------------------------|--|
| ID | Attributs (étiquette, visibilité, w , h , c_x , c_y , $\hat{w}(\frac{px}{fr})$, $\hat{h}(\frac{px}{fr})$, $\hat{c}_x(\frac{px}{fr})$, $\hat{c}_y(\frac{px}{fr})$, $h_o(A)$) |
| A | (1, 1, 41, 104, 94, 207, 3, 0, 15, 0, ) |
| B | (2, 1, 57, 150, 151, 220, 4, 1, 12, 4, ) |
| C | (3, 1, 94, 197, 235, 233, 4, 2, 8, 8, ) |
| D | (4, 1, 40, 81, 326, 212, 1, 1, 0, 10, ) |
| E | (5, 1, 105, 253, 505, 308, 0, 0, 0, 0, ) |
| <i>Attributs des arêtes</i> | |
| ID | Mesure d'adjacence adj (avec $\tau = 0.1$) |
| e(A,B) | 0.62 |
| e(A,C) | 0.43 |
| e(B,C) | 0.58 |
| e(B,D) | 0.32 |
| e(C,D) | 0.27 |
| e(D,E) | 0.12 |

TABLE 7.3 – Valeurs des attributs du graphe représenté dans l'exemple de la Figure 7.4.

Dans notre cas, cela signifie que nous prenons le graphe état le plus proche du graphe observation.

7.4 CONCLUSIONS

Le suivi d'objet est, peut-être, de toutes les composantes d'un système de vidéo surveillance, l'étape la plus difficile et pour cette raison celle qui a eu le plus d'attention de la part des chercheurs. Beaucoup de propositions ont été faites, mais le problème reste encore ouvert parce que les méthodes sont, soit efficaces mais coûteuses, soit rapides mais pas très performantes.

Nous avons proposé une méthode qui, en exploitant les liens hiérarchiques entre différentes résolutions des représentations d'un objet, devient suffisamment efficace pour gérer les correspondances multiples et donc les occlusions. Cette méthode a été ma première tentative d'utiliser une représentation structurelle pour résoudre le problème de suivi. Mon opinion est que les relations (spatiales, temporelles, etc.) entre les objets sont une information importante pour la compréhension de la scène et donc pour la résolution des problèmes liés au suivi.

Pour cette raison, mon travail actuel continue dans cette direction et vise toujours à introduire des données structurées dans les méthodes de suivi. C'est pourquoi, comme les méthodes basées sur les filtres particuliers se sont avérées efficaces, ma nouvelle activité de recherche vise à introduire une représentation par graphes dans les filtres particuliers. Ces travaux sont actuellement en cours et correspondent à la méthodologie expliquée dans la section précédente.

APPLICATIONS

8

SOMMAIRE

| | |
|--|----|
| 8.1 ANALYSE DES TRAJECTOIRES | 91 |
| 8.2 ESTIMATION DE FOULE | 93 |
| 8.3 RÉ-IDENTIFICATION DE PERSONNES | 95 |
| 8.4 PERSPECTIVES | 98 |

GRÂCE aux étapes et techniques décrites préalablement, un système d'analyse de vidéos est capable d'extraire les trajectoires d'objets, alors exploitables dans nombreuses applications. Les trajectoires peuvent, par exemple, être utilisées pour modéliser des actions et donc le système pourrait avoir l'objectif de déclencher une alarme lorsqu'une action particulière est détectée. Dans le cadre des applications de sécurité, les actions pourraient être un arrêt prolongé dans une zone suspecte, une personne qui laisse un bagage abandonné, etc. Les trajectoires pourraient être aussi analysées pour apprendre des activités récurrentes (sur la base de la connaissance de l'environnement) et par conséquent permettre de détecter des trajectoires "anormales" par rapport aux modèles appris. Enfin les trajectoires, convenablement stockées dans des bases de données spécifiques, pourraient servir pour la recherche et l'extraction de séquences vidéo contenant des événements d'intérêt.

Nous avons abordé la thématique de l'analyse des trajectoires grâce à une collaboration avec Luc Brun de l'ENSICAEN. Dans ce contexte, j'ai participé à l'encadrement d'une thèse (Saggese (2013)) en co-tutelle entre le laboratoire de Caen et mon laboratoire en Italie. Cette thèse, dont une synthèse est présentée dans la Section 8.1, aborde le problème de la détection des trajectoires "anormales" par rapport à un ensemble de trajectoires normales précédemment apprises. La thèse développe une technique d'apprentissage des trajectoires normales, basée sur le clustering des trajectoires représentées par des chaînes de caractères en utilisant des noyaux de chaînes de caractères pour la mesure de similarité utilisée dans le clustering.

Mais, les systèmes d'analyse de vidéos ne s'arrêtent pas à l'analyse des trajectoires. Nous en avons abordé deux en particulier : l'estimation de foule qui vise à estimer le nombre de personnes présentes dans une zone donnée ; la ré-identification de personnes qui consiste à établir une

correspondance appropriée entre les observations d'une même personne qui pourrait apparaître, disparaître et réapparaître à des moments différents ou dans des vidéos provenant de différentes caméras.

Pour ces deux problématiques nous allons présenter un état de l'art et décrire nos propositions dans les Sections 8.2 (estimation de foule) et 8.3 (ré-identification de personnes). Ces contributions ont été publiées dans [Conte et al. (2010d;b; 2013a)] et dans [Brun et al. (2011b), Mahboubi et al. (2013b; 2014)].

Nous concluons le chapitre avec certaines perspectives dans la Section 8.4.

8.1 ANALYSE DES TRAJECTOIRES

Dans une vidéo, une activité peut être caractérisée par l'ensemble des mouvements des objets de la scène, à savoir les trajectoires. Une activité peut impliquer un seul objet (par exemple, un véhicule traversant la route dans le mauvais sens) ou un groupe d'objets (par exemple, deux personnes qui se battent) [Morris et Trivedi (2008)]. Nous proposons un algorithme qui, une fois les modèles des trajectoires appris, est en mesure de calculer à quel point une nouvelle trajectoire est semblable à un modèle, et donc de reconnaître les trajectoires anormales (pas assez semblables aux modèles donnés).

Le processus d'apprentissage des modèles d'activités exige que chaque trajectoire soit bien représentée et que les modèles typiques soient calculés. Dans ce but, une stratégie de clustering doit être définie.

Il est clair que la représentation choisie influe fortement sur la façon de coder la similarité entre les trajectoires. Par exemple, dans [Acampora et al. (2012)] les données brutes sont utilisées et la similitude est évaluée en utilisant un réseau neuronal. Le principal inconvénient de cette méthode est que de grandes quantités de données sont nécessaires au cours de la phase d'apprentissage. Dans [Hsieh et al. (2006), Gaur et al. (2009)] les trajectoires sont représentées par une séquence de symboles et les approches d'apprentissage statistique sont exploitées. Dans [Hsieh et al. (2006)] la similitude entre les trajectoires est évaluée en utilisant la distance d'édition, tandis que dans [Gaur et al. (2009)] les auteurs utilisent la technique du *Dynamic Time Warping* (DTW). Le principal problème des techniques mentionnées réside dans le fait que, bien qu'elles soient capable de calculer une distance, elles ne sont pas en mesure de définir facilement une métrique. Dans [Xiang et Gong (2008)] les trajectoires sont modélisées et classées dans une matrice de transition d'état et une chaîne de Markov cachée (HMM) est utilisée. En général, une grande quantité de données est nécessaire pour éviter le problème de "overfitting" pendant l'étape d'apprentissage de la HMM.

Une limitation commune aux approches mentionnées réside dans la perte d'informations importantes causée par les représentations choisies (c'est ce qui arrive par exemple avec les représentations vectorielles). D'autres représentations sont efficaces mais la mesure de similarité n'induit pas une métrique ce qui rend impossible l'utilisation des outils traditionnels d'apprentissage et classification.

Afin de surmonter les limites des approches existantes, nous proposons de représenter les trajectoires par des chaînes de caractères et une métrique de similarité basée sur les noyaux. De ce fait, nous ne perdons pas d'informations utiles dans la représentation et nous exploitons les avantages de l'utilisation des noyaux : avoir une métrique qui nous permet d'utiliser les outils de classification statistique. En outre, un nouvel algorithme de clustering basé sur les noyaux a été défini.

La méthode est basée sur les étapes suivantes (Figure 8.1) :

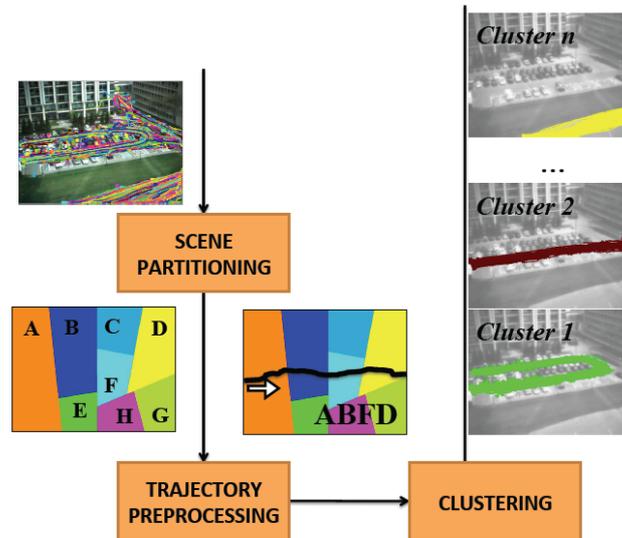


FIGURE 8.1 – Le schéma proposé pour l'analyse des trajectoires ([Saggese (2013)]).

Partitionnement de la scène Afin de réduire la quantité de données à gérer, la scène est divisée en zones. Les trajectoires sont représentées en prenant en compte les zones dans lesquelles les points de la trajectoire sont présents. Les zones sont définies de façon dynamique en fonction de la densité de points des trajectoires extraites, de manière à ce que les parties de l'image où la plupart des trajectoires se trouvent, soient représentées par un grand nombre de zones.

Représentation des Trajectoires Chaque trajectoire est représentée par une séquence de symboles, à savoir une chaîne de caractères. Chaque symbole contient les caractéristiques nécessaires (position, zones traversées, etc.) pour discriminer deux trajectoires différentes. La similarité entre deux trajectoires est évaluée en utilisant un procédé basé sur un noyau.

Clustering Etant donné le noyau, un nouvel algorithme de clustering est appliqué afin d'en extraire des groupes (et des prototypes) de trajectoires à l'intérieur de la scène. Chaque modèle correspond à un chemin typique. L'algorithme est une version modifiée du clustering proposé dans [Brun et Trémeau (2002)]; la modification se situe dans la mesure de similarité qui est substituée par le noyau.

Les détails des différentes étapes sont décrits dans la thèse [Saggese (2013)].

Une fois que les prototypes des trajectoires normales ont été trouvés, les trajectoires anormales peuvent être facilement détectées par comparaison (à l'aide des noyaux) avec les prototypes.

Pour les expérimentations nous avons utilisés deux bases de données publiques de trajectoires : MIT Trajectory dataset¹ (MIT), Edinburgh Informatics Forum Pedestrian Database² (EDH). Les bases de données ont été annotées manuellement par des experts, afin de définir les trajectoires normales et anormales.

L'index utilisé a été le C-index [Hubert et Schultz (1976)] pour évaluer

1. <http://www.ee.cuhk.edu.hk/~xgwang/MITtrajsingle.html>

2. <http://homepages.inf.ed.ac.uk/rbf/FORUMTRACKING/>

| Méthode | MIT | EDH |
|--------------------------|------|------|
| Approche proposée | 0.07 | 0.08 |
| K-means | 0.29 | 0.31 |
| Global K-means | 0.23 | 0.27 |
| PCA | 0.26 | 0.28 |

TABLE 8.1 – Résultats du clustering (C-index) pour l'application d'analyse de trajectoires.

l'efficacité du clustering (et donc l'efficacité de l'apprentissage) par rapport à d'autres méthodes bien connues. La Table 8.1 montre les résultats qui confirment la validité de l'approche proposée. D'autres résultats sont décrits dans [Saggese (2013)].

8.2 ESTIMATION DE FOULE

La connaissance du nombre de personnes présentes dans une zone est une question importante dans le cadre des applications d'analyse vidéo et un nombre croissant d'articles sur ce sujet ont été proposés récemment. Parmi les applications concernées par cette question, on peut citer la surveillance vidéo (un nombre excessif de personnes dans une zone peut constituer un danger pour la sécurité ou la sûreté), le marketing (par exemple, déterminer comment les clients sont répartis dans un grand centre commercial), etc.

Les occlusions provoquent généralement une sous-estimation importante dans le processus de comptage. Un autre point essentiel est la présence inévitable de distorsions de perspective : les personnes loin de la caméra semblent petites dans les images, tandis que celles qui sont proches sont beaucoup plus grandes. Par conséquent, les méthodes de comptage doivent faire face à ces questions de perspective.

La littérature présente deux approches différentes. L'approche directe, qui repose sur la détection individuelle des personnes, en utilisant des algorithmes de segmentation et de détection d'objets adéquats. D'autre part, dans l'approche indirecte le nombre de personnes est estimé par la mesure de la présence de caractéristiques convenablement définies qui ne nécessitent pas la détection séparée de chaque personne dans la scène. Ces caractéristiques sont alors mises en relation avec le nombre de personnes. Parmi les exemples récents et bien connus de l'approche directe on peut citer Brostow et Cipolla (2006), Zhao et al. (2008). L'approche directe présente l'avantage que la détection de personnes est souvent déjà effectuée pour d'autres applications ; la méthode est efficace tant que les personnes sont correctement segmentées, et que les occlusions ne gênent pas beaucoup, etc. Mais, comme indiqué dans les chapitres précédents, la détection de personnes est une tâche complexe. Bien que la phase de détection soit nécessaire dans les applications basées sur la reconstruction des trajectoires des objets, dans le cas de l'estimation de foule cette étape peut être évitée grâce aux approches indirectes. Les approches indirectes sont basées sur l'extraction de mesures convenablement définies et posent le problème de trouver une correspondance exacte entre ces mesures et le nombre de personnes. Malgré leur simplicité, les approches dans cette

catégorie montrent des performances prometteuses (Albiol et al. (2009), Chan et al. (2008)).

La dernière catégorie semble être la plus prometteuse, mais parfois imprécise à cause de la perspective.

L'approche que nous proposons se base sur la subdivision de la scène entière en bandes horizontales pour prendre en compte la perspective. En outre, un procédé entièrement automatisé pour la construction de ces bandes a été proposé.

L'idée sous-jacente de notre approche est que chaque personne peut être représentée par un petit nombre de points saillants. Ces points peuvent être situés aux bords de la silhouette et en correspondance de certaines autres parties qui présentent des discontinuités élevées (les yeux, la bouche, le nez, les vêtements, etc.). Sous l'hypothèse que les personnes sont les seuls objets en mouvement dans la scène, le nombre total de personnes peut être estimé à partir de l'ensemble des points saillants détectés :

$$P = \omega \cdot N \quad (8.1)$$

où P est une estimation du nombre de personnes dans la scène, N est le nombre de points saillants détectés dans l'image, tandis que ω est une constante de proportionnalité.

L'hypothèse de représenter une personne par un nombre constant de points saillants est trop simpliste car elle ne tient pas compte de plusieurs questions comme, pour n'en nommer que quelques-unes, l'optique de la caméra, la position de la personne dans la scène par rapport à la caméra, les obstacles entre les personnes, etc.

Ainsi, l'équation est reformulée comme suit :

$$P = \sum_{i=1}^N \omega(p_i) \quad (8.2)$$

Cette formulation permet facilement d'inclure des informations locales pour chaque point saillant ; les hypothèses sous-jacentes sont que l'information locale est suffisante pour le problème de comptage, et que cette information peut être combinée de manière additive.

Afin d'atteindre le but, notre méthode est basée sur les idées suivantes :

- La fonction $\omega(\cdot)$ ne dépend que de la distance entre le point saillant et le plan d'image :

$$P = \sum_{i=1}^N \omega(d(p_i)) \quad (8.3)$$

- où $d(p_i)$ est la distance du point de la caméra ;
- $\omega(\cdot)$ est modélisée comme une fonction constante par morceaux, en ayant la scène divisée en bandes horizontales et en utilisant une valeur constante pour chaque bande ;
- l'apprentissage de $\omega(\cdot)$ est effectuée à l'aide d'une procédure automatique, qui ne nécessite que l'acquisition d'une courte séquence vidéo ; la construction n'exige pas que les personnes dans la vidéo soient segmentées manuellement.

| Vidéo (caméra) | Albiol Albiol et al. (2009) | | Méthode proposée | |
|-----------------|-----------------------------|--------|------------------|-------|
| | MAE | MRE | MAE | MRE |
| S1.L1.13-57 (1) | 2.80 | 12.6% | 1.36 | 6.8% |
| S1.L1.13-59 (1) | 3.86 | 24.9% | 2.55 | 16.3% |
| S1.L2.14-06 (1) | 5.14 | 26.1% | 5.40 | 20.8% |
| S1.L3.14-17 (1) | 2.64 | 14.0% | 2.81 | 15.1% |
| S1.L1.13-57 (2) | 29.45 | 106.0% | 4.45 | 15.1% |
| S1.L2.14-06 (2) | 32.24 | 122.5% | 12.17 | 30.7% |
| S1.L2.14-31 (2) | 34.09 | 99.7% | 7.55 | 23.6% |
| S3.MF.12-43 (2) | 12.34 | 311.9% | 1.64 | 35.2% |

TABLE 8.2 – Estimation du nombre de personnes par l’algorithme de Albiol et la méthode proposée.

Les détails sont donnés dans l’article [Conte et al. (2013a)].

Un exemple de la division de l’image en bandes est représenté dans la Figure 8.2. Les bandes correspondent à la hauteur d’une personne en fonction de la perspective.



FIGURE 8.2 – Subdivision de la vidéo en bandes

La conférence PETS2009 a organisé un concours pour l’estimation de la densité de personnes dans les vidéos ; nous avons utilisé les séquences de benchmark proposées lors de cette conférence. La Table 8.2 présente la comparaison entre notre système et le meilleur système résultant de la compétition PETS. A partir des résultats reportés dans le tableau Table 8.2, il est évident que la méthode proposée (dans presque tous les cas) surpasse la technique de Albiol [Albiol et al. (2009)] à l’égard des deux indices de performance MAE (valeur absolue de l’erreur de comptage) et MRE (valeur relative de l’erreur de comptage). Cet aspect est plus évident si l’on se réfère aux résultats obtenus sur la deuxième caméra où les problèmes de perspective sont plus évidents.

8.3 RÉ-IDENTIFICATION DE PERSONNES

Le problème de la ré-identification de personnes consiste à trouver une correspondance entre les personnes filmées à travers différentes caméra ou dans différents instants de temps. Plusieurs applications utilisant une seule caméra peuvent bénéficier d’une information induite par la

ré-identification. L'une des principales applications est la détection de la flânerie. La flânerie se réfère à la présence prolongée de personnes dans une zone. Ce comportement est intéressant pour détecter, par exemple, les mendiants dans les coins de rue, ou les trafiquants de drogue dans les gares routières, etc. À côté de cela, les informations sur ces ré-occurrences sont très importantes dans la configuration multi-caméra. Ces systèmes de surveillance visent à identifier une personne d'intérêt dans des endroits différents avec des caméras dont les points de vue ne se chevauchent pas.

Le problème de la ré-identification a été étudié depuis dix ans environ. Un premier groupe Bird et al. (2005), Gandhi et Trivedi (2006) traite ce problème en définissant une signature unique qui condense un ensemble d'images d'une même personne ; la ré-identification est ensuite effectuée en utilisant une mesure de similarité entre les signatures et un seuil pour estimer si une nouvelle image de personne représente une nouvelle entrée ou une personne déjà vue. Un deuxième groupe ([Bazzani et al. (2010), TruongCong et al. (2010)]) traite la ré-identification par le biais d'une représentation basée sur une seule image de la personne. Chaque représentation correspond à un point dans un espace. Ensuite, une classification est effectuée en regroupant ces points à l'aide d'une SVM ou d'un calcul de corrélation.

Nous avons proposé une méthodologie qui appartient au deuxième groupe. L'idée principale est de capturer les informations essentielles d'une personne qui soient invariantes à différentes transformations (changement d'échelle, de pose, etc.). Une représentation structurelle pourrait donc être efficace à cet effet. Pour cette raison nous avons proposé un certain nombre de représentations pour les personnes et nous avons utilisé les noyaux (voir chapitre 4) dans la phase de classification par le biais d'outils tels que SVM, K-PCA, etc.

Une première approche [Brun et al. (2011b)] se base sur l'idée que l'apparence d'une personne peut être divisée en trois ou quatre régions principales pour lesquelles la couleur, la surface relative et d'autres caractéristiques ne changent pas sur plusieurs images de la même personne. La méthode est donc basée sur une représentation de la personne par un graphe en se basant sur une segmentation de l'image correspondante. Nous partons du masque de la personne détectée dans la phase de détection des objets en mouvement (voir chapitre 6) ; la partie de l'image courante de la vidéo qui correspond au masque de la personne est segmentée et finalement cette segmentation sert à construire un graphe d'adjacence (Region Adjacency Graph). La Figure 8.3 montre un exemple de la représentation d'une personne par graphe d'adjacence. Le noyau utilisé pour la comparaison entre les graphes afin de détecter si deux images représentent la même personne ou pas, est le noyau basé sur la distance d'édition entre graphes décrit dans le chapitre 4.

Dans une deuxième approche [Mahboubi et al. (2013b)], la représentation se base sur l'idée que pour une personne les relations spatiales entre les points caractéristiques de son apparence, ne changent pas dans le mouvement. Donc nous avons proposé une représentation du graphe par le

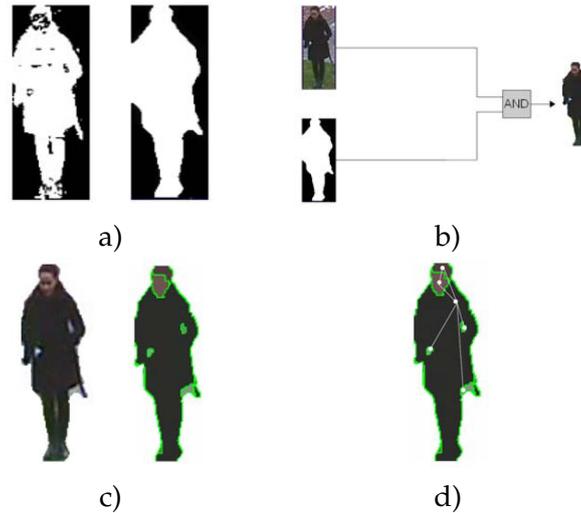


FIGURE 8.3 – Phases de la construction du RAG : a) Application d'un opérateur morphologique au masque de foreground ; b) Extraction de l'apparence de la personne ; c) Segmentation de l'image ; d) le RAG final.

biais des points SIFT [Lowe (1999)] : sur le masque de foreground de la personne à représenter on calcule les points SIFT ; ensuite nous les codons par un graphe $G = (V, E, w)$ où V correspond à l'ensemble des points SIFT, E est l'ensemble des arêtes (deux sommets sont connectés entre eux si l'un fait partie des k plus proches voisins de l'autre et vice versa) et w est une fonction de poids définie sur V et elle est définie comme l'échelle de l'apparence du sommet correspondant. Le noyau défini pour cette approche est basé sur la notion de sac de motifs que nous avons illustré dans le chapitre 4 sur le plongement.

Finalement, une troisième approche [Mahboubi et al. (2014)] est basée sur une représentation des personnes par chaîne de caractères. L'idée principale de cette approche se base sur le fait que (encore plus que pour les contours) le squelette d'une personne (à l'exclusion des bras et des jambes) est plutôt stable dans le temps. Nous partons toujours du masque de foreground de la personne. Ensuite, la construction de la chaîne de caractères est faite en trois étapes : tout d'abord, on applique un détecteur de front (opérateur de Deriche) sur le masque binaire ; la deuxième étape comporte la construction d'une courbe qui caractérise approximativement le centre de gravité de la personne ; cette courbe est créée à partir des bords extérieurs de la personnes en calculant, pour chaque ligne horizontale de l'image, la position moyenne entre les coordonnées x des bords considérés ; la dernière étape est un lissage de la courbe calculée à l'étape précédente. La Figure 8.4 montre les étapes de la construction de la chaîne de caractères représentant une personne. Pour cette approche, le noyau utilisé est un noyau de chaîne de caractère connu, appelé "Global Alignment Kernel" [Cuturi (2011)].

Dans la Table 8.3 une comparaison entre les méthodes précédentes est

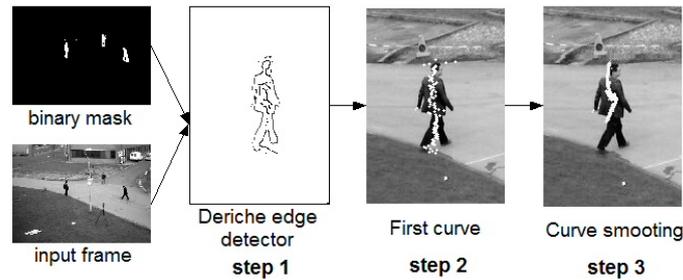


FIGURE 8.4 – Représentation par chaîne de caractères d’une image de personne.

| Méthode | Taux de classification |
|--|------------------------|
| Noyau basé sur les RAG | 70% |
| Noyau basé sur le points SIFT | 65% |
| Noyau basé sur les chaînes de caractères | 55% |

TABLE 8.3 – Taux de classification des différentes méthodes de ré-identification sur la base de données PETS2010.

montrée. On a utilisé la base de données publique PETS [The PETS 2010 dataset (2010)]. On peut remarquer que le taux de classification (c’est-à-dire le pourcentage de fois pour lequel l’algorithme ré-identifie correctement une personne) pour les trois méthodes est relativement élevé. Donc on peut faire l’hypothèse que les différents noyaux définis sont assez efficaces pour la solution du problème de la ré-identification de personnes.

Il serait nécessaire d’effectuer une comparaison détaillée des propositions illustrées avec d’autres approches de référence dans la thématique de la ré-identification pour valider l’efficacité de ces propositions.

8.4 PERSPECTIVES

Une première perspective autour des applications d’analyse vidéo est de traduire les techniques proposées à d’autres cadres applicatifs, comme par exemple le domaine de l’imagerie médicale, dans lequel les vidéos sont de plus en plus disponibles grâce aux nouveaux instruments d’acquisition.

Je me suis occupé d’analyse d’images médicales à l’occasion d’un co-encadrement d’une thèse ([Tufano (2007)]) sur la segmentation d’images par résonance magnétique, par la mise au point d’améliorations de l’algorithme de Level Set. En ce moment, je co-encadre une thèse ([Hadjerci (Soutenance prévue en 2016)]) dans le cadre d’un projet régional (DANIEAL - Région Centre) sur la conception d’un outil de détection automatique des blocs nerveux pour guider le praticien vers ces blocs afin qu’il puisse injecter le produit analgésique. A cet effet, nous envisageons d’utiliser les techniques d’analyse des trajectoires (vu dans ce chapitre) pour détecter des trajectoires “anormales” qui correspondraient à des gestes erronés de la part du médecin.

Dans le cadre du comptage de personnes, une perspective intéressante peut être d'exploiter la détection du foreground pour améliorer la performance de la solution. En effet nous avons montré qu'à la sortie de la phase de détection, le masque de détection est assez fiable ; l'idée serait donc de calculer les caractéristiques (comme les points saillants utilisés dans notre proposition) seulement sur les pixels du premier-plan, afin de prendre en compte seulement les informations utiles dans l'image, en réduisant ainsi le bruit.

Dans le contexte de la ré-identification, les définitions de noyaux jusqu'ici développées sont un peu rigides : pour avoir une similarité élevée entre deux graphes, il faut que ces derniers soient très similaires entre eux dans leur structure. Pourtant lors de la comparaison de deux images d'une personne, l'effet de petites perturbations dans l'apparence, en raison des variations de luminosité ou des changements de pose, n'est pas négligeable. Par conséquent, les graphes représentant les deux images peuvent être très différents même s'ils représentent la même personne. Donc, la définition d'une similitude entre deux graphes devrait offrir ce type de souplesse. On envisage donc la définition d'un nouveau noyau qui, dans la mesure de similarité entre deux graphes, tiendrait aussi compte de ce besoin d'"appariement inexact".

CONCLUSION GÉNÉRALE

Dans ce mémoire nous avons résumé nos contributions à la RF structurée et à l'analyse vidéo. Le lecteur n'aura pas manqué de repérer le potentiel amené par l'utilisation de données structurées dans la RF. Nous avons vu, en effet, que les graphes (ou d'autres structures comme les cartes) peuvent être utilisés pour représenter des données et des interactions complexes et, à l'aide d'outils de manipulations appropriés, il est possible de concevoir des méthodes qui améliorent les performances dans divers domaines de la RF. Durant mon parcours scientifique, j'ai montré en particulier comment les graphes et les données structurées peuvent être efficaces pour la résolution des problèmes ouverts en vidéo surveillance. Tous nos travaux de recherche ont été réalisés en collaboration avec plusieurs chercheurs français (J.M. Jolion, L. Brun, A. Mahboubi, J.Y. Ramel, G. Damiand, D. Villemin) et étrangers (M. Vento, P. Foggia, C. Sansone, G. Percannella, J. Gibert, M.M. Luqman) tandis que beaucoup de travaux liés au domaine applicatif de la vidéo surveillance ont été réalisés dans le cadre de collaborations industrielles.

Nous nous sommes intéressés tout d'abord à l'appariement de graphes. Nous avons montré qu'il n'existe pas encore un algorithme qui soit le plus efficace parmi toutes les propositions. Nous avons prouvé, au contraire, que les performances d'un algorithme dépendent de la typologie des graphes utilisés. Avec cette contribution nous avons permis aux chercheurs d'avoir un cadre de référence lorsqu'ils doivent choisir le bon algorithme d'appariement pour un problème spécifique en connaissant la typologie des graphes représentant leurs données.

Ensuite nous nous sommes intéressés à un outil qui permet d'exploiter la richesse de l'information contenue dans les graphes avec la robustesse des outils de la RF statistique : le plongement de graphes. Avec le plongement de graphes nous transformons les graphes dans un espace vectoriel tout en essayant de garder la plupart des informations structurées contenues dans les données originelles. Cette transformation peut être explicite (Graph Embedding) ou implicite (Noyaux de graphes). Nous avons proposé la définition de plusieurs noyaux de graphes basés sur la distance d'édition entre graphes et sur la notion de sac de motifs.

Enfin, nous avons présenté en détail l'architecture d'un système de vidéo surveillance, les techniques proposées dans l'état de l'art et nos propositions qui visent à résoudre certains problèmes ouverts dans le domaine. La plupart des propositions sont basées sur les méthodes de RF structurées présentées dans la première partie.

Si des perspectives ont déjà été données dans les conclusions des différents chapitres, nous revenons sur certaines d'entre elles pour préciser

les collaborations et les projets à venir qui me tiennent particulièrement à cœur.

Je compte continuer la collaboration avec L. Brun sur les noyaux de graphes appliqués à la ré-identification de personnes. Dans ce cadre, on a déjà envisagé un nouveau noyau, basé sur l'édition des chemins d'un graphe, comme nous l'avons cité dans le chapitre 8.

De même, je désirerais continuer la collaboration avec I. Bloch et S. Dubuisson sur le suivi d'objets par filtre particulière. La conception d'un filtre particulière basé sur une représentation par graphe me semble très originale et intéressante.

Je crois que les cartes généralisées permettent le développement de nouvelles (et efficaces) méthodes dans la RF. J'espère aussi pouvoir continuer la collaboration avec G. Damiand sur ce sujet pour lequel il reste des idées intéressantes à développer : la construction incrémentale des cartes pour un traitement en ligne des vidéos, ou l'application des cartes à la reconnaissance d'actions ou dans le suivi d'objets ne sont que quelques exemples.

Toutes ces perspectives concernent des applications de vidéo surveillance. Pour cette raison j'ai proposé, cette année, dans le cadre de l'appel à projet ANR 2015, un projet Jeune Chercheur sur le thème de la sécurité des personnes à travers le développement d'un système de vidéo surveillance intelligent. Dans le cadre de ce projet je compte développer les perspectives décrites.

Enfin, grâce à la participation (en tant que co-responsable) à un projet industriel (DOD) dans mon équipe de recherche actuelle, je commence à travailler dans la thématique de l'analyse de documents. Le traitement et l'analyse de documents, en particulier la reconnaissance de caractères et de symboles ont été un des premiers domaines d'applications des graphes en RF, et de nombreuses possibilités restent encore ouvertes dans ce domaine. Cela devrait me permettre de poursuivre et étendre mes contributions au domaine de la RF structurelle dans un contexte autre que la vidéo surveillance et avec l'aide des futurs doctorants que j'aurais l'opportunité d'encadrer.

BIBLIOGRAPHIE

- G. Acampora, P. Foggia, A. Saggese, et M. Vento. Combining neural networks and fuzzy systems for human behavior understanding. Dans *IEEE Ninth International Conference on Advanced Video and Signal-Based Surveillance (AVSS)*, 2012. (Cité page 91.)
- A. Albiol, M.J. Silla, A. Albiol, et J.M. Mossi. Video analysis using corner motion statistics. Dans *IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*, 2009. (Cité pages 94 et 95.)
- P.S. Aleksandrov. *Elementary concepts of topology*. Dover Publications Inc., New York, 1961. (Cité page 67.)
- Najib Ben Aoun, Mahmoud Mejdoub, et Chokri Ben Amar. Graph-based approach for human action recognition using spatio-temporal features. *Journal of Visual Communication and Image Representation*, 25-2 :1047–3203, 2014. (Cité page 46.)
- D. Arsic, A. Lyutskanov, G. Rigoll, et B. Kwolek. Multi camera person tracking applying a graph-cuts based foreground segmentation in a homography framework. Dans *Twelfth IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETSWinter)*, 2009. (Cité page 83.)
- E. Balas et C. S. Yu. Finding a maximum clique in an arbitrary graph. *SIAM J. Computing*, 15(4) :1054–1068, 1986. (Cité page 20.)
- L. Bazzani, M. Cristani, A. Perina, M. Farenzena, et V. : Murino. Multiple-shot person re-identification by hpe signature. Dans *Proceedings of 20th International Conference on Pattern Recognition, ICPR*, 2010. (Cité page 96.)
- J. Berclaz, F. Fleuret, et P. Fua. Multiple object tracking using low linear programming. Dans *Twelfth IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS-Winter)*, 2009. (Cité page 83.)
- J. Berclaz, F. Fleuret, E. Turetken, et P. Fua. Multiple object tracking using k-shortest paths optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33-9 :1806–1819, 2011. (Cité page 77.)
- K. Bhuvaneshwari et H. Abdul Rauf. Edgelet based human detection and tracking by combined segmentation and soft decision. Dans *International Conference on Control, Automation, Communication and Energy Conservation*, 2009. (Cité page 78.)
- N. Bird, O. Masoud, N. Papanikolopoulos, et A. Isaacs. Detection of loitering individuals in public transportation areas. *IEEE Transactions on Intelligent Transportation Systems*, 6-2 :167–177, 2005. (Cité page 96.)

- C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006. (Cité pages 1 et 84.)
- B. Bouchon-Meunier, M. Rifqi, et S. Bothorel. Towards general measures of comparison of objects. *Fuzzy Sets and Systems*, 84-2 :143–153, 1996. (Cité page 86.)
- G.J. Brostow et R. Cipolla. Unsupervised bayesian detection of independent motion in crowds. Dans *IEEE Conf. on Computer Vision and Pattern Recognition*, 2006. (Cité page 93.)
- L. Brun, D. Conte, P. Foggia, et Vento M. A graph-kernel method for re-identification. Dans M. Kamel et A. Campilho (Eds.), éditeurs, *Proceedings of the 8th International Conference on Image Analysis and Recognition (ICIAR 2011)*, volume 6753 de *Lecture Notes in Computer Science*, pages 173–182, 2011a. (Cité pages 31 et 39.)
- L. Brun, D. Conte, P. Foggia, et Vento M. People re-identification by graph kernels methods. Dans Xiaoyi Jiang, Miquel Ferrer, et Andrea Torsello (Eds.), éditeurs, *Proceedings of the 8th IAPR-TC-15 International Workshop, GbRPR 2011*, volume 6658 de *Lecture Notes in Computer Science*, pages 285–294, 2011b. (Cité pages 31, 90 et 96.)
- L. Brun, D. Conte, P. Foggia, M. Vento, et D. Villemin. Symbolic learning vs. graph kernels : An experimental comparison in a chemical application. Dans *Proceedings of the 14th Conference on Advances in DataBases and Information Systems (ADBIS 2010)*, pages 31–40, 2010. (Cité page 31.)
- L. Brun et A. Trémeau. Digital color imaging handbook. Electrical and Applied Signal Processing, Chapitre Color quantization, pages 589–637. CRC Press, 2002. (Cité page 92.)
- H. Bunke et M. Vento. Benchmarking of graph matching algorithms. Dans *Proceedings of the 2nd Workshop on Graph-based Representations*, 1999. (Cité page 24.)
- Kaspar Riesenand Horst Bunke. Graph classification and clustering based on vector space embedding. *International Journal on Pattern Recognition and Artificial Intelligence*, - :-, 2009. (Cité page 34.)
- T. Caelli et S. Kosinov. An eigenspace projection clustering method for inexact graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26-4 :515–519, 2004. (Cité page 21.)
- A. Cavallaro et T. Ebrahimi. Change detection based on color edges. Dans *The 2001 IEEE Intern. Symp. on Circuits and Systems, ISCAS*, 2001. (Cité page 54.)
- A.B. Chan, Z.S.J. Liang, et N. Vasconcelos. Privacy preserving crowd monitoring : counting people without people models or tracking. Dans *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008. (Cité page 94.)

- Z. Chen, T. Ellis, et S. A. Velastin. "vehicle detection, tracking and classification in urban traffic. Dans *Proceedings of the 15th International IEEE Conference on Intelligent Transportation Systems*, 2012. (Cité page 77.)
- Fan R. K. Chung. *Spectral Graph Theory*. American Mathematical Society, 1997. (Cité page 37.)
- D. Comaniciu, V. Ramesh, et P. Meer. Real-time tracking of non-rigid objects using mean shift. Dans *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 2000. (Cité page 77.)
- D. Conte et G. Damiand. Remove noise in video with 3d topological maps. Dans *Workshops on Statistical Techniques in Pattern Recognition and Structural and Syntactic Pattern Recognition (SSPR)*, 2014. (Cité pages 51, 55 et 64.)
- D. Conte, P. Foggia, J. M. Jolion, et M. Vento. A graph-based, multi-resolution algorithm for tracking objects in presence of occlusions. Dans L. Brun et M.Vento (Eds.), éditeurs, *Proceedings of the 5th Workshop on Graph-based Representation in Pattern Recognition (GbRPR2005)*, volume 3434 de *Lecture Notes in Computer Science*, pages 193–202, 2005a. (Cité page 14.)
- D. Conte, P. Foggia, J.M. Jolion, et M. Vento. A graph-based, multi-resolution algorithm for tracking objects in presence of occlusions. *Pattern Recognition Journal*, 39-4 :562–572, 2006. (Cité pages 14, 26, 75 et 79.)
- D. Conte, P. Foggia, G. Percannella, A. Saggese, et M. Vento. An ensemble of rejecting classifiers for anomaly detection of audio events. Dans *Proceedings of the 9th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS 2012)*, pages 76–81, 2012a. (Cité page xiii.)
- D. Conte, P. Foggia, G. Percannella, F. Tufano, et M. Vento. An algorithm for recovering camouflage errors on moving people. Dans Edwin R. Hancock, Richard C. Wilson, Terry Windeatt, Ilkay Ulusoy, et Francisco Escolano (Eds.), éditeurs, *Joint IAPR International Workshop on Structural, Syntactic, and Statistical Pattern Recognition*, volume 6218 de *Lecture Notes in Computer Science*, pages 365–374, 2010a. (Cité pages 55 et 70.)
- D. Conte, P. Foggia, G. Percannella, F. Tufano, et M. Vento. Counting moving people in videos by salient points detection. Dans *20th International Conference on Pattern Recognition (ICPR2010)*, pages 1743–1746, 2010b. (Cité page 90.)
- D. Conte, P. Foggia, G. Percannella, F. Tufano, et M. Vento. An experimental evaluation of foreground detection algorithms in real scenes. *EURASIP Journal on Advances in Signal Processing*, 2010 :373941 :11, 2010c. (Cité page 54.)
- D. Conte, P. Foggia, G. Percannella, F. Tufano, et M. Vento. A method for counting people in crowded scenes. Dans *Seventh IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS 2010)*, pages 225–232, 2010d. (Cité page 90.)

- D. Conte, P. Foggia, G. Percannella, et M. Vento. Performance evaluation of a people tracking system on pets2009 database. Dans *Seventh IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS 2010)*, pages 119–126, 2010e. (Cité page 83.)
- D. Conte, P. Foggia, G. Percannella, et M. Vento. Removing object reflections in videos by global optimization. *IEEE Transactions on Circuits and Systems for Video Technology*, 22-11 :1623–1633, 2012b. (Cité page 55.)
- D. Conte, P. Foggia, G. Percannella, et M. Vento. Counting moving persons in crowded scenes. *Machine Vision and Applications*, 24 - 5 :1029–1042, 2013a. (Cité pages 90 et 95.)
- D. Conte, P. Foggia, M. Petretta, F. Tufano, et M. Vento. Evaluation and improvements of a real-time background subtraction method. Dans M. Kamel et A. Campilho (Eds.), éditeurs, *Proceedings of the International Conference on Image Analysis and Recognition (ICIAR 2005)*, volume 3656 de *Lecture Notes in Computer Science*, pages 1234–1241, 2005b. (Cité page 70.)
- D. Conte, P. Foggia, M. Petretta, F. Tufano, et M. Vento. Meeting the application requirements of intelligent video surveillance systems in moving object detection. Dans S. Singh, M. Singh, C. Apte, et P. Perner (Eds.), éditeurs, *Proceedings of the International Conference on Advances in Pattern Recognition (ICAPR 2005)*, volume 3687 de *Lecture Notes in Computer Science*, pages 653–662, 2005c. (Cité page 57.)
- D. Conte, P. Foggia, C. Sansone, et M. Vento. Thirty years of graph matching in pattern recognition. *International Journal of Pattern Recognition and Artificial Intelligence*, 18-3 :265–298, 2004. (Cité page 13.)
- D. Conte, P. Foggia, F. Tufano, et M. Vento. Evaluation and improvements of the level set method for rm images segmentation. Dans *Proceedings of International Conference on Computer Vision Theory and Applications (VISAPP 2009)*, pages 210–215, 2009. (Cité page xiii.)
- D. Conte, P. Foggia, F. Tufano, et M. Vento. An enhanced level set algorithm for wrist bone segmentation. Dans Pei-Gee Ho (Ed.), éditeur, *Image Segmentation*, pages 293–308. InTech, 2011. (Cité page xiii.)
- D. Conte, P. Foggia, et M. Vento. Challenging complexity of maximum common subgraph detection algorithms : A performance analysis of three algorithms on a wide database of graphs. *Journal of Graph Algorithms and Applications*, 11-1 :99–143, 2007. (Cité pages 14, 23, 24 et 25.)
- D. Conte, C. Guidobaldi, et C. Sansone. A comparison of three maximum common subgraph algorithms on a large database of labeled graphs. Dans E. Hancock et M. Vento (Eds.), éditeurs, *Proceedings of the International Workshop on Graph-based Representation in Pattern Recognition (GbR 2003)*, volume 2726 de *Lecture Notes in Computer Science*, pages 130–141, 2003. (Cité page 14.)

- D. Conte, J.-Y. Ramel, N. Sidère, M. M. Luqman, B. Gauzère, J. Gibert, L. Brun, et M. Vento. A comparison of explicit and implicit graph embedding methods for pattern recognition. Dans *Proceedings of the 9th IAPR-TC15 workshop on Graph-based Representation in Pattern Recognition (Gbr 2013)*, pages 81–90, 2013b. (Cité pages 31, 38 et 40.)
- D. Conte et F. Tufano. Segmenting wrist bone tissue : Enhancements to level set algorithm. Dans *Proceedings of the International Workshop on Medical Measurements and Applications (MeMeA 2009)*, pages 177–182, 2009. (Cité page xiii.)
- D. J. Cook, N. Manocha, et L. B. Holder. Using a graph-based data mining system to perform web search. *International Journal of Pattern Recognition and Artificial Intelligence*, 17-5 :705–720, 2003. (Cité page 20.)
- L. Cordella, P. Foggia, C. Sansone, et M. Vento. Performance evaluation of the vf graph matching algorithm. Dans *Proceedings of the 10th International Conference on Image Analysis and Processing*, 1999. (Cité page 24.)
- L. P. Cordella, P. Foggia, C. Sansone, et M. Vento. An improved algorithm for matching large graphs. Dans *Workshop on Graph-Based Representations in Pattern Recognition*, 2001. (Cité page 20.)
- M. Cuturi. Icml. Chapitre Fast global alignment kernels, pages 929–936. Omnipress, 2011. (Cité page 97.)
- C. Dai, Y. Zheng, et X. Li. Pedestrian detection and tracking in infrared imagery using shape and appearance. *Computer Vision and Image Understanding*, 106-23 :288–299, 2007. (Cité page 77.)
- G. Damiand. Topological model for 3d image representation : Definition and incremental extraction algorithm. *CVIU*, 109(3) :260–289, March 2008. URL <http://dx.doi.org/10.1016/j.cviu.2007.09.007>. (Cité pages 66 et 68.)
- G. Damiand. Combinatorial maps. Dans *CGAL User and Reference Manual*. CGAL Editorial Board, 3,9 édition, 2010a. URL <http://www.cgal.org/Pkg/CombinatorialMaps>. (Cité page 66.)
- G. Damiand. *Contributions aux Cartes Combinatoires et Cartes Généralisées : Simplification, Modèles, Invariants Topologiques et Applications*. PhD thesis, Mémoire de l’Habilitation à Diriger des Recherches en Informatique - Université Claude Bernard Lyon 1, 2010b. (Cité pages 65 et 66.)
- DARPA. Section i, video surveillance and monitoring. Dans *Proc. of the DARPA Image Understanding Workshop*, 1998. (Cité page 45.)
- T. Darrell, G. G. Gordon, M. Harville, et J. Woodfill. Integrated person tracking using stereo, color and pattern detection. *International Journal of Computer Vision*, 37 :175–185, 2000. (Cité page 45.)
- C. de la Higuera, J.-C. Janodet, E. Samuel, G. Damiand, et Christine Solnon. Polynomial algorithms for open plane graph and subgraph isomorphisms. *Theoretical Computer Science*, 498 :76–99, 2013. (Cité page 26.)

- R. Di Lascio, P. Foggia, A. Saggese, et M. Vento. Tracking interacting objects in complex situations by using contextual reasoning. Dans *Proceedings of International Conference on Computer Vision Theory and Applications (VISAPP)*, 2012. (Cité page 83.)
- A. Doucet, N. De Freitas, et N. Gordon. *Sequential Monte Carlo Methods in Practice*. Springer, 2001. (Cité pages 84 et 85.)
- A. Doucet, B. Vo, C. Andrieu, et M. Davy. Particle filtering for multi-target tracking and sensor management. Dans *Fifth International Conference on Information Fusion*, 2002. (Cité page 78.)
- A. Dupas et G. Damiand. Region merging with topological control. *Discrete Applied Mathematics*, 157(16) :3435–3446, August 2009. URL <http://dx.doi.org/10.1016/j.dam.2009.04.005>. (Cité pages 35, 67 et 68.)
- F. X. Dupé et L. Brun. Tree covering within a graph kernel framework for shape classification. Dans *Proceedings of the XV International Conference on Image Analysis and Processing*, 2009. (Cité page 35.)
- P. J. Durand, R. Pasari, J. W. Baker, et C. Tsai. An efficient algorithm for similarity analysis of molecules. *Internet Journal of Chemistry*, 2 :-, 1999. (Cité pages 17 et 20.)
- A. Dziech, J. Bialas, A. Glowacz, P. Korus, M. Leszczuk, A. Matiolalski, et R. Baran. Overview of recent advances in cctv processing chain in the induct and insigma projects. Dans *Eighth International Conference on Availability, Reliability and Security*, 2013. (Cité page 45.)
- J. M. Ferryman, S. J. Maybank, et A. D. Worrall. Visual surveillance for moving vehicles. *International Journal of Computer Vision*, 37-2 :187–197, 2000. (Cité page 45.)
- James Ferryman. Pets 2014. Dans *IEEE International Workshop on Performance Evaluation of Tracking and Surveillance - Korea*, 2013. (Cité page 45.)
- P. Foggia, R. Genna, et M. Vento. Symbolic vs connectionist learning : an experimental comparison in a structured domain. *IEEE Transaction on Knowledge and Data Engineering*, 13-2 :176–195, 2001a. (Cité page 37.)
- P. Foggia, C. Sansone, et M. Vento. A performance comparison of five algorithms for graph isomorphism. Dans *Proceedings of the 3rd IAPR TC-15 Workshop on Graph-based Representations in Pattern Recognition (GbR)*, 2001b. (Cité page 22.)
- Y. Freund et R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55 :119–139, 1997. (Cité page 78.)
- T. Gandhi et M.M. Trivedi. Panoramic appearance map (pam) for multi-camera based person re-identification. Dans *IEEE International Conference on Video and Signal Based Surveillance, AVSS*, 2006. (Cité page 96.)

- U. Gaur, B. Song, et A. Roy-Chowdhury. Query-based retrieval of complex activities using strings of motion-words. Dans *Workshop on Motion and Video Computing*, 2009. (Cité page 91.)
- B. Gauzère, L. Brun, et D. Villemain. Graph kernels based on relevant patterns and cycle information for chemoinformatics. Dans *Proceedings of the International Conference on Pattern Recognition*, 2012. (Cité page 39.)
- J. Gibert, E. Valveny, et H. Bunke. Graph embedding in vector spaces by node attribute statistics. *Pattern Recognition*, 45(9) :3072–3083, 2012. (Cité page 39.)
- C. Gomila et F. Meyer. Graph-based object tracking. Dans *Proceedings of the International Conference on Image Processing, ICIP*, 2003. (Cité page 46.)
- S. Gosselin, G. Damiand, et C. Solnon. Efficient search of combinatorial maps using signatures. *Theoretical Computer Science*, 412(15) :1392–1405, 2011. (Cité page 41.)
- O. Hadjerci, A. Hafiane, P. Makris, D. Conte, P. Vieyres, et A. Delbos. Nerve detection in ultrasound images using median gabor binary pattern. Dans *Proceedings of the 11th International Conference on Image Analysis and Recognition (ICIAR 2014)*, 2014. (Cité page xii.)
- Oussama Hadjerci. *Détection de Régions d'Intérêt dans des Images Echographiques*. PhD thesis, INSA Val de Loire - Bourges, Soutenance prévue en 2016. (Cité page 98.)
- I. Haritaoglu, D. Harwood, et L. S. Davis. W₄ : Real-time surveillance of people and their activities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22 :809–830, 2000. (Cité pages 45, 53, 54 et 77.)
- P. E. Hart, N. J. Nilsson, et B. Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, 2 :100–107, 1968. (Cité page 19.)
- David Haussler. Convolution kernels on discrete structures. Rapport technique, Department of Computer Science, University of California at Santa Cruz, 1999. (Cité page 35.)
- B.K.P. Horn et B.G. Schunck. Determining optical flow. *Artificial Intelligence*, 17 :185–204, 1981. (Cité page 53.)
- J.-W. Hsieh, S.-L. Yu, et Y.-S. Chen. Motion-based video retrieval by trajectory matching. *IEEE Transactions on Circuits and Systems for Video Technology*, 16-3 :396–409, 2006. (Cité page 91.)
- W. Hu, W. Hu, N. Xie, et S. Maybank. Unsupervised active learning based on hierarchical graph-theoretic clustering. *IEEE Transactions on Systems, Man, and Cybernetics, Part B : Cybernetics*, 39-5 :1147–1161, 2009. (Cité page 45.)
- L. Hubert et J. Schultz. Quadratic assignment as a general data analysis strategy. *British Journal of Mathematical and Statistical Psychology*, 29-2 : 190–241, 1976. (Cité page 92.)

- A. Inokuchi, T. Washio, et H. Motoda. An apriori-based algorithm for mining frequent substructures from graph data. Dans *4th European Conference on Principles of Data Mining and Knowledge Discovery*, 2000. (Cité page 33.)
- M. Isard et A. Blake. Condensation - conditional density propagation for visual tracking. *International Journal on Computer Vision*, 28-1 :5–28, 1998. (Cité page 78.)
- R. Jain et H. Nage. On the analysis of accumulative difference pictures from image sequences of real world scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1-2 :206–214, 1979. (Cité page 53.)
- Z. Jiang, D. Q. Huynh, W. Moran, et S. Challa. Tracking pedestrians using smoothed colour histograms in an interacting multiple model framework. Dans *International Conference on Image Processing*, 2011. (Cité page 77.)
- JIVP. Special issue on advanced video-based surveillance. *Journal of Image Video Processing*, 2011 :1687–5176, 2011. (Cité page 45.)
- J.M. Jolion et A. Montanvert. The adaptive pyramid : a framework for 2d image analysis. *CVGIP : Image Understanding*, 55 :339–348, 1992. (Cité pages 26 et 80.)
- S. Jouili et S. Tabbone. Graph embedding using constant shift embedding. Dans *Proceedings of the International Conference on Pattern Recognition*, 2010. (Cité page 34.)
- D. Justice et A. Hero. A binary linear programming formulation of the graph edit distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28 (8) :1200–1214, 2006. (Cité page 21.)
- P. Kaewtrakulpong et R. Bowden. An improved adaptive background mixture model for realtime tracking with shadow detection. Dans *Proc. 2nd European Workshop on Advanced Video Based Surveillance Systems, AVBS01, Video Based Surveillance Systems : Computer Vision and Distributed Processing*, 2001. (Cité page 57.)
- Hisashi Kashima, Koji Tsuda, et Akihiro Inokuchi. Marginalized kernel between labeled graphs. Dans *In Proc. of the Twentieth International conference on Machine Learning*, 2003. (Cité pages 35 et 36.)
- R. Kasturi, D. Goldgof, P. Soundararajan, V. Manohar, J. Garofolo, R. Bowers, M. Boonstra, V. Korzhova, et J. Zhang. Framework for performance evaluation of face, text, and vehicle detection and tracking in video : Data, metrics, and protocol. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31 :319–336, 2009. (Cité page 82.)
- C. Kim et J. Hwang. A fast and robust moving object segmentation in video sequences. Dans *International Conference on Image Processing*, 1999. (Cité page 53.)

- Janusz Konrad et Marek Domanski. Avss. Dans *10th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, 2013. (Cité page 45.)
- V.A. Kovalevsky. Finite topology as applied to image analysis. *CVGIP*, 46 : 141–161, 1989. (Cité page 67.)
- Walter G. Kropatsch. Equivalent contraction kernels to build dual irregular pyramids. *Advances in Computer Science, Advances in Computer Vision*, - :99–107, 1997. (Cité page 26.)
- P. LeBodic, P. Hèroux, S. Adam, et Y. Lecourtier. An integer linear program for substitution-tolerant subgraph isomorphism and its use for symbol spotting in technical drawings. *Pattern Recognition*, 45 (12) :4214–4224, 2012. (Cité page 21.)
- Liyuan Li, Weimin Huang, Irene Y. H. Gu, et Qi Tian. Foreground object detection from videos containing complex background. Dans *Proceedings of the eleventh ACM international conference on Multimedia*, 2003. (Cité page 57.)
- P. Lienhardt. Topological models for boundary representation : a comparison with n-dimensional generalized maps. *Computer Aided Design*, 23(1) :59–82, 1991. (Cité page 64.)
- P. Lienhardt. N-dimensional generalized combinatorial maps and cellular quasi-manifolds. *Int. Journal of Computational Geometry and Applications*, 4(3) :275–324, 1994. (Cité page 66.)
- D.-T. Lin et K.-Y. Huang. "collaborative pedestrian tracking and data fusion with multiple cameras. *IEEE Transactions on Information Forensics and Security*, 6-4 :1432–1444, 2011. (Cité page 77.)
- D. G. Lowe. Object recognition from local scale-invariant features. Dans *Proceedings of the International Conference on Computer Vision*, 1999. (Cité page 97.)
- Artur Loza, Lyudmila Mihaylova, David Bull, et Nishan Canagarajah. Structural similarity-based object tracking in multimodality surveillance videos. *Machine Vision and Applications*, 20-2 :71–83, 2009. (Cité page 46.)
- Bin Luo, Richard C. Wilson, et Edwin R. Hancock. Spectral embedding of graphs. *Pattern Recognition*, 36(10) :2213–2230, 2003. (Cité page 33.)
- M.M. Luqman, J.Y. Ramel, J. Lladòs, et T. Brouard. Fuzzy multilevel graph embedding. *Pattern Recognition*, 46(2) :551–565, 2013. (Cité pages 38 et 39.)
- D.J.C. MacKay. Introduction to monte carlo methods. *NATO Advanced Study Institute on Learning in Graphical Models*, - :175–204., 1998. (Cité page 78.)
- L. Maddalena et A. Petrosino. A self-organizing approach to background subtraction for visual surveillance applications. *IEEE Transactions on Image Processing*, 17-7 :1168–1177, 2008. (Cité pages 57 et 58.)

- Pierre Mahé et Jean-Philippe Vert. Graph kernels based on tree patterns for molecules. *Machine Learning*, 75(1) :3–35, 2008. (Cité page 35.)
- A. Mahboubi, L. Brun, D. Conte, P. Foggia, et M. Vento. Ré-identification de personnes par modèle de noyaux de graphe. Dans *24ème Edition du colloque GRETSI*, 2013a. (Cité pages 31 et 40.)
- A. Mahboubi, L. Brun, D. Conte, P. Foggia, et M. Vento. Tracking system with re-identification using a graph kernels approach. Dans *Proceedings of the 15th International Conference on Computer Analysis of Images and Patterns*, pages 401–408, 2013b. (Cité pages 31, 40, 90 et 96.)
- A. Mahboubi, L. Brun, D. Conte, P. Foggia, et M. Vento. Tracking system with re-identification using a rgb string kernel. Dans *Workshops on Statistical Techniques in Pattern Recognition and Structural and Syntactic Pattern Recognition (SSPR)*, 2014. (Cité pages 31, 40, 90 et 97.)
- L. Marcenaro, M. Ferrari, L. Marchesotti, et C.S. Regazzon. Multiple object tracking under heavy occlusions by using kalman filters based on shape matching. Dans *IEEE International Conference on Image Processing*, 2002. (Cité page 54.)
- T. Matsuyama. Cooperative distributed vision. Dans *Proc. of the DARPA Image Understanding Workshop*, 1998. (Cité page 45.)
- M. Bertozzi et A. Broggi. Gold : A parallel real-time stereo vision system for generic obstacle and lane detection. *IEEE Transactions on Image Processing*, 7-1 :62–81, 1998. (Cité pages 48 et 49.)
- J. McGregor. Backtrack search algorithms and the maximal common subgraph problem. *Software Practice and Experience*, 12 :23–34, 1982. (Cité page 20.)
- Peter Meer. Stochastic image pyramids. *CVGIP*, 45 :269–294, 1989. (Cité page 26.)
- B. Morris et M. Trivedi. A survey of vision-based trajectory learning and analysis for surveillance. *IEEE Transactions on Circuits and Systems for Video Technology*, 18-8 :1114–1127, 2008. (Cité page 91.)
- J.R. Munkres. *Elements of Algebraic Topology*. Perseus Books, 1984. (Cité page 67.)
- M Neuhaus et H. Bunke. *Bridging the Gap between Graph Edit Distance and Kernel Machines*. World Scientific Publishing, 2007. (Cité page 35.)
- M. Neuhaus, K. Riesen, et H. Bunke. Fast suboptimal algorithms for the computation of graph edit distance. Dans *Joint International Workshops on Statistical Techniques in Pattern Recognition and Structural and Syntactic Pattern Recognition (SSPR)*, 2006. (Cité page 17.)
- K. Okuma, A. Taleghani, N. de Freitas, J. Little, et D. Lowe. A boosted particle filter : multitarget detection and tracking. Dans *European Conference on Computer Vision*, 2004. (Cité page 78.)

- B. Ozer, W. Wolf, et A. N. Akansu. A graph based object description for information retrieval in digital image and video libraries. Dans *Proc. of the IEEE Workshop Content-Based Access of Image and Video Libraries*, 1999. (Cité page 45.)
- T.M. Paixao, A. Graciano, R.M. Cesar, et R. Hirata. A backmapping approach for graph-based object tracking. Dans *XXI Brazilian Symposium on Computer Graphics and Image Processing*, 2008. (Cité page 46.)
- S. Pellegrini, A. Ess, K. Schindler, et L. van Gool. You'll never walk alone : Modeling social behavior for multi-target tracking. Dans *IEEE 12th International Conference on Computer Vision*, 2009. (Cité page 77.)
- A. Prati, I. Mikic, M.M. Trivedi, et R. Cucchiara. Detecting moving shadows : algorithms and evaluation. *IEEE Trans. on PAMI*, 25(7) :918–923, 2003. (Cité page 58.)
- H. Qiu et E. R. Hancock. Graph matching and clustering using spectral partitions. *Pattern Recognition*, 39-1 :22–34, 2006. (Cité page 21.)
- R. C. Wilson R. Myers et E. R. Hancock. Bayesian graph edit distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22 :628–635, 2000. (Cité page 21.)
- Peng Ren, R.C. Wilson, et E.R. Hancock. Graph characterization via ihara coefficients. *IEEE Transactions on Neural Networks*, 22-2 :233–245, 2011. (Cité page 33.)
- A. Robles-Kelly et E. Hancock. Graph edit distance from spectral seriation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27-3 :365–378, 2005. (Cité page 21.)
- A. Rosenfeld. Adjacency in digital pictures. *Information and Control*, 26(1) :24–33, 1974. (Cité page 66.)
- M. Rota et M. Thonnat. Video sequence interpretation for visual surveillance. Dans *Proc. of the 3rd IEEE Int. Workshop on Visual Surveillance*, 2000. (Cité page 45.)
- A. Saggese. *Detecting and indexing moving objects for Behavior Analysis by Video and Audio Interpretation*. PhD thesis, ENSICAEN et Université de Salerno, 2013. (Cité pages 89, 92 et 93.)
- M. Salotti et N. Laachfoubi. Topographic graph matching for shift estimation. Dans *Proc. of the 3rd IAPR-TC15 Workshop Graph-Based Representations in Pattern Recognition*, 2001. (Cité page 45.)
- A. Sanfeliu et K. Fu. A distance measure between attributed relational graphs for pattern recognition. *IEEE Transactions on Systems, Man, and Cybernetics (Part B)*, 13 :353–363, 1983. (Cité page 17.)
- K. Shearer, H. Bunke, et S. Venkatesh. Video indexing and similarity retrieval by largest common subgraph detection using decision trees. *Pattern Recognition*, 34 :1075–1091, 2001. (Cité page 45.)

- N. Shervashidze, S. V.N. Vishwanathan, T. H. Petri, K. Mehlhorn, et K. M. Borgwardt. Efficient graphlet kernels for large graph comparison. Dans *Twelfth International Conference on Artificial Intelligence and Statistics*, 2009. (Cité page 35.)
- N. Sidere, P. Heroux, et J.-Y. Ramel. A vectorial representation for the indexation of structural informations. Dans *Proceedings of the International Conference on Statistical + Structural and Syntactical Pattern Recognition*, 2008. (Cité page 33.)
- N. Sidère, P. Héroux, et J.Y. Ramel. Vector representation of graphs : Application to the classification of symbols and letters. Dans *International Conference on Document Analysis and Recognition*, 2009. (Cité pages 33, 38 et 39.)
- A.J. Smola et R.I. Kondor. Kernels and regularization on graphs. Dans *XV Annual Conference on Learning Theory*, 2003. (Cité page 37.)
- Christine Solnon. Alldifferent-based filtering for subgraph isomorphism. *Artificial Intelligence*, 174(12-13) :850–864, 2010. (Cité page 20.)
- C. Stauffer et W.E.L. Grimson. Learning patterns of activity using real-time tracking. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 22-8 :747–757, 2000. (Cité page 54.)
- Florian Steinke et Bernhard Schölkopf. Kernels, regularization and differential equations. *Pattern Recognition*, 41(11) :3271 – 3286, 2008. ISSN 0031-3203. (Cité page 36.)
- Frédéric Suard, Alain Rakotomamonjy, et Abdelaziz Bensrhair. Kernel on bag of paths for measuring similarity of shapes. Dans *European Symposium on Artificial Neural Networks*, Bruges-Belgique, April 2007. (Cité page 36.)
- Murtaza Taj, Emilio Maggio, et Andrea Cavallaro. Multi-feature graph-based object tracking. Dans *First International Evaluation Workshop on Classification of Events, Activities and Relationships, CLEAR*, 2006. (Cité page 46.)
- H. Hai Tao, Sawhney, et R. Kumar. Object tracking with bayesian estimation of dynamic layer representations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24-1 :75–b89, 2002. (Cité page 78.)
- The PETS 2010 dataset. <http://www.cvg.rdg.ac.uk/pets2010/a.html>. Dans *PETS 2010 Conference*, 2010. (Cité pages 69, 82 et 98.)
- TIFS. Special issue on intelligent video surveillance for public security & personal privacy. *IEEE Transactions on Information Forensics and Security*, - :-, 2013. (Cité page 45.)
- K. Toyama, J. Krumm, B. Brumitt, et B. Meyers. Wallflower : Principles and practice of background maintenance. Dans *Seventh IEEE International Conference on Computer Vision*, 1999. (Cité page 54.)

- D.N. TruongCong, L. Khoudour, C. Achard, C. Meurie, et O. Lezoray. People reidentification by spectral classification of silhouettes. *Signal Processing*, 90 :2362–2374, 2010. (Cité page 96.)
- W. H. Tsai et K. S. Fu. Subgraph error-correcting isomorphisms for syntactic pattern recognition. *IEEE Transactions on Systems Man and Cybernetics*, 13 :48–61, 1983. (Cité page 20.)
- Francesco Tufano. *Segmentazione di immagini biomediche con approcci ciechi ed approcci basati su modello*. PhD thesis, Université de Salerno (Italie), 2007. (Cité page 98.)
- J. Kittler W. J. Christmas et M. Petrou. Structural matching in computer vision using probabilistic relaxation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17 :749–764, 1995. (Cité page 21.)
- Nicolas Widynski, Séverine Dubuisson, et Isabelle Bloch. Fuzzy spatial constraints and ranked partitioned sampling approach for multiple object tracking. *Computer Vision and Image Understanding*, 116 :1076–1094, 2012. (Cité page 78.)
- R. Wilson, E. Hancock, et B. Luo. Pattern vectors from algebraic graph theory. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27 :1112–1124, 2005. (Cité page 33.)
- C. Wolf et J.-M. Jolion. Model based text detection in images and videos : a learning approach. Rapport technique, LIRIS INSA de Lyon, 2004. (Cité page 70.)
- C. R. Wren, A. Azarbayejani, T. Darrel, et A. P. Pentland. Pfunder : Real-time tracking of the human body. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 19-7 :780–785, 1997. (Cité page 54.)
- T. Xiang et S. Gong. Incremental and adaptive abnormal behaviour detection. *Computer Vision and Image Understanding*, 111-1 :59–73, 2008. (Cité page 91.)
- J. Zhang, L. L. Presti, et S. Sclaroff. Online multi-person tracking by tracker hierarchy. Dans *Proceedings of the 9th AVSS Conference*, 2012. (Cité page 77.)
- T. Zhao, R. Nevatia, et B. : Wu. Segmentation and tracking of multiple humans in crowded environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30-7 :1198–1211, 2008. (Cité page 93.)

Titre Contributions à la reconnaissance de formes structurelle et à l'analyse vidéo

Résumé Le résumé en français (\approx 1000 caractères)

Mots-clés Reconnaissance de Formes, Graphes, Appariement de Graphes, Plongement de Graphes, Noyaux de Graphes, Cartes Généralisées, Analyse Vidéo, Vidéo Surveillance, Suivi d'objets, Ré-identification de Personnes

Title Structural Pattern Recognition and Video Analysis Contributions

Abstract Le résumé en anglais (\approx 1000 caractères)

Keywords Pattern Recognition, Graphs, Graphs Matching, Graph Embedding, Graph Kernels, Generalized Maps, Video Analysis, Video Surveillance, Object Tracking, People Re-identification