Vision "fruste" revisitée : contribution à la vision dynamique des systèmes

Samia Bouchafa

UNIVERSITÉ PARIS SUD XI

École Doctorale STITS

INSTITUT D'ELECTRONIQUE FONDAMENTALE

HABILITATION À DIRIGER DES RECHERCHES

présentée par :

Samia BOUCHAFA

Spécialité : Traitement des images ; Vision par ordinateur

Vision "fruste" revisitée : Contribution à la vision dynamique des systèmes

Soutenue le novembre 2011, devant le jury constitué de :

Rapporteurs :	Virginio CANTONI	Professeur, Università degli Studi di Pavia
	Jean DEVARS	Professeur, Université Paris VI
	Jack-Gérard POSTAIRE	Professeur émérite, Université Lille I
Examinateurs :	Philippe BOLON	Professeur, Université de Savoie
	Michel DEVY	Directeur de recherche, CNRS LAAS, Toulouse
	Philippe TARROUX	Professeur, Ecole Normale Supérieure
	Bertrand ZAVIDOVIQUE	Professeur, Université Paris Sud-XI

Remerciements

Je remercie chaleureusement les rapporteurs de ce manuscrit : Virginio Cantoni, Professeur à l'Université de Pavie dont je n'oublierais pas l'accueil chaleureux lors de ma visite dans son laboratoire. J'ai eu l'occasion de lire ses ouvrages passionnants qui m'ont particulièrement marqués. Spécialiste de la transformée de Hough, j'ai naturellement pensé qu'il serait intéressé par les approches cumulatives proposées ici. Jean Devars, Professeur à l'Université Paris VI qui a toujours suivi avec intérêt mon parcours. Son recul scientifique et ses conseil avisés m'ont toujours éclairée. Jack-Gérard Postaire, Professeur Emérite à l'Université Lille I qui avait déjà eu la tâche d'assurer la fonction de rapporteur de ma thèse en 1998. Ayant clairement perçu l'intérêt des approches basées sur les lignes de niveaux, il m'a encouragé à poursuivre dans ce sens.

J'exprime également tous mes remerciements aux examinateurs : Philippe Bolon, Professeur à l'Université de Savoie, Michel Devy, Directeur de recherche au LAAS à Toulouse et Philippe Tarroux, Professeur à l'Ecole Normale Supérieure, directeur adjoint du LIMSI.

Je remercie vivement Bertrand Zavidovique, Professeur à l'Université Paris Sud-XI avec qui j'ai toujours un immense plaisir à travailler. Je n'oublierais pas son *brainstorming* légendaire lors des séances de travail passionnantes avec lui.

Je remercie également mes collègues d'AXIS tous si attachants : Franck Bimbard, Samir Bouaziz, Flavien Delgehier, Abdelhafid Elouardi, Michel Fan, Michèle Gouiffès, Bruno Larnaudie, Lionel Lacassagne, François Le Coat, Alain Lambert, Sylvie Le-Hégarat, Alain Mérigot, Roger Reynaud, Marius Vasiliu mais aussi mes collègues du LSS Anthony Busson, Hugues Mounier et Véronique Vèque ainsi que mes collègues du LIVIC. Merci en particulier à Didier Aubert qui m'a toujours fait confiance et avec qui j'ai eu la joie de collaborer régulièrement.

Mes pensées vont également à mes trois collègues partis à la retraite Elisabeth Bouyssy, Claude Arcile et Claudine Falcetta dont je n'oublierai pas les qualités humaines.

Je remercie toutes les personnes avec lesquelles j'ai collaboré au cours de ces années : étudiants, doctorants, collègues chercheurs et enseignants-chercheurs.

Je tiens à remercier également ma famille et ma belle-famille pour leur soutien constant.

Aux trois soleils de ma vie : mon époux Olivier, mes fils Elias et Diwan...

Table des matières

Organisation du mémoire

Ce mémoire décrit mes activités de recherche au sein de l'Institut d'Electronique Fondamentale de l'Université Paris Sud-XI. Il est organisé en trois parties.

- La première, qui constitue le curriculum vitae détaillé, est consacrée à la description de mes activités d'enseignement et de recherche. Les rubriques classiques d'un CV, dans l'ordre recommandé par l'Université Paris Sud-XI, s'y trouvent. J'ai choisi de présenter dans cette partie l'ensemble de mes activités de recherche par ordre chronologique, ce qui permet d'apprécier l'évolution des thématiques traitées. Cet ordre pratique, nécessaire dans un premier temps, sera complété dans la partie suivante afin de mettre en évidence la cohérence des approches proposées.

- C'est pourquoi la seconde partie de ce mémoire reprend les principaux thèmes de manière détaillée mais non exhaustive afin de révéler la cohérence de mes choix méthodologiques. Elle se divise en deux chapitres principaux : l'un consacré aux primitives issues des lignes de niveaux que nous avons proposées, l'autre aux processus de décision cumulatifs. Le second chapitre est lui même découpé en 3 sections, chacune dédiée à une manière différente d'appréhender l'accumulation.

- La dernière partie reproduit cinq publications représentatives de l'ensemble de mes travaux.

Première partie

Bilan des activités de recherche et d'enseignement

Chapitre 1

Curriculum vitae détaillé

Samia BOUCHAFA-BRUNEAU
Née le 16/07/1971 à Alger, nationalité Française
Mariée, 2 enfants (nés en 2004 et 2006)

Adresse professionnelle :
Université Paris Sud-XI
Bât. 220, 91405, Orsay Cedex
☎ 01 69 15 40 07

Adresse personnelle :
4 allée du japon
91300, Massy
☎ 01 60 11 03 36

✉ samia.bouchafa@u-psud.fr

1.1 Situation actuelle

Depuis septembre 1999 : Maître de conférences de la classe normale.
Section CNU : 61 (Génie Informatique, Automatique et Traitement du Signal).
Rattachée au département de Physique, Université Paris Sud-XI.
Actuellement en Délégation au LIVIC (IFSTTAR) depuis le 1er oct. 2011 jusqu'au 30 sept. 2012.

Laboratoire : Institut d'Electronique Fondamentale (IEF), CNRS UMR 8622.

Département : Architectures, Contrôle, Communication, Image, Systèmes (ACCIS).

Équipe : Vision, Image, Systèmes Autonomes (VISA).

1.2 Titres universitaires

14 décembre 1998 : Thèse de doctorat, Université Paris VI.

- TITRE : Détection du mouvement insensible aux variations de contraste. Application à la détection de mouvements de foules anormaux dans le métro par traitement d'images.
- LABORATOIRE : Institut National de Recherche sur les Transports et leur Sécurité (INRETS [1]), Département Analyse et Régulation du trafic (DART [2]).
- MENTION : "Très honorable avec les félicitations du jury".
- COMPOSITION DU JURY :

Didier Demigny	Rapporteur, Professeur, ENSEA Cergy
Jack-Gérard Postaire	Rapporteur, Professeur, Université Lille I
Jean-Marc Blosseville	Examinateur, Directeur de recherche, INRETS
Maurice Milgram	Président du jury, Professeur, Université Paris VI
Didier Aubert	Examinateur, Chargé de recherche, INRETS
Jean Devars	Directeur de thèse, Professeur, Université Paris VI

1994 : DEA Robotique, Université Paris VI. Stage de DEA : "Traqueur 3D par traitement d'images pour une application à la Réalité Virtuelle" au Laboratoire de Robotique de Paris sous la direction de Philippe Coiffet. Participation à la coupe de France de Robotique, équipe du LRP, La Ferté Bernard.

1993 : Diplôme d'ingénieur en informatique, Institut National d'Informatique [3], Alger. Cursus d'ingénieur en 5 ans. Option "systèmes informatiques". Mention "Très Bien". Concours national d'entrée après le Baccalauréat, classée 3ème sur 1300 candidats.

1.3 Autres éléments du parcours

1999 : Ingénieur de développement en Traitement d'images. Société CITILOG (Carrefour Intelligent Traitement d'images Logiciels). Développements dans le cadre du projet "détection de présence" sur les ponts mobiles par traitement d'images. Collaboration avec VNF (Voies Navigables de France).

2008 : Congé pour Recherche ou Conversion Thématique, 1 semestre, au LIVIC (Laboratoire sur les Interactions Véhicules-Infrastructure-Conducteur), unité mixte INRETS/LCPC. J'ai choisi la période de septembre 2008 à janvier 2009.

1. Au 1er janvier 2011, l'INRETS et le LCPC ont fusionné pour donner naissance à l'IFSTTAR Institut Français des Sciences et Technologies des Transports, de l'Aménagement et des Réseaux.

2. Une partie des chercheurs du DART s'est constitué (en 2000) en une nouvelle unité mixte INRETS/LCPC : le LIVIC Laboratoire sur les Interactions Véhicules-Infrastructure-Conducteurs.

3. Actuellement : ESI (Ecole nationale Supérieure d'Informatique)

1.4 Activités d'enseignement

J'ai été recrutée en 1999 sur un poste d'enseignant chercheur dont le profil en enseignement était "Traitement des images". Un manque d'intervenants dans cette discipline justifiait la création de ce poste aussi bien au niveau de la Maîtrise EEA (M1 IST), du DEA SETI (M2 R SETI), du DESS Systèmes Electroniques (M2 Pro GSE) que de la FIUPSO (Polytech. Paris Sud-XI). Les sections ci-dessous montrent le point de vue adopté durant ces années pour assurer mes enseignements, mon implication dans des filières/domaines ayant souffert de manière ponctuelle ou récurrente de manques d'enseignants, ainsi que mon engagement constant à travers le montage de nouveaux enseignements ou l'encadrement d'étudiants.

Investissement dans les montage de nouveaux Travaux Pratiques :

Dès mon recrutement, je me suis particulièrement investie dans le montage de nouveaux Travaux pratiques et Travaux dirigés, les cours étant assurés par le Professeur B. Zavidovique. A cette occasion, je n'ai pas hésité à enrichir la bibliothèque d'algorithmes de traitement des images de Khoros en programmant de nouveaux modules pour les besoins en enseignement : algorithmes d'analyse d'images (contours, régions, texture, etc.), de détection et d'estimation de mouvement (détection par mise à jour de référence, estimation de flot optique, etc.) ou de vision 3D (mise en correspondance stéréoscopique, calibrage, etc.) récents. J'ai donc monté de nouveaux travaux pratiques et géré leur développement au fil des années en fonction de l'évolution des environnements de programmation : nous sommes passés successivement de Khoros sous linux à Visiquest sous windows ; actuellement, nous exportons nos routines sous ImageJ. Durant cette période, j'ai été responsable de la salle TP de Traitement d'image de l'UFR ainsi que celle de la FIUPSO [4]. Cette responsabilité nécessitait, de ce fait, la gestion informatique quotidienne des machines en réseaux sous Linux. De la même manière, j'ai été amenée à monter de nouveaux TP de synthèse d'images (Tracé de primitives, clipping, projections, rendu couleur/texture, z-buffer) et à réaliser des librairies d'algorithmes nécessaires aux étudiants pendant les séances de travaux pratiques.

Point de vue adopté pour l'enseignement du Traitement d'images :

J'ai été amenée à intervenir progressivement pour des cours magistraux de Traitement des images en particulier pour le Master 1, Master Professionnel et Master Recherche (voir tableau récapitulatif). Le traitement des images a été envisagé sous trois points de vues différents, selon les filières d'enseignement et la formation initiale des étudiants. Le premier est en lien avec le traitement du signal (M1 IST). L'image est considérée comme un signal 2D et les traitement classiques sur le signal sont étendus aux traitement des images 2D voire 3D (en ajoutant la dimension temporelle). Le second point de vue aborde l'image au sens d'une modalité dont il faut extraire des information pertinentes en vue de traitement de plus haut niveaux (M1 IST, M2 Pro GSE, IFIPS). Nous abordons alors l'analyse d'images proprement dite : détection de contours, points d'intérêts, texture, mouvement, couleur.

4. Formation d'Ingénieurs de l'Université Paris Sud Orsay. Actuellement : Polytech. Paris Sud-XI

Enfin, le troisième point de vue concerne les aspects de plus haut niveau tels que la vision 3D (stéréovision, analyse du mouvement 3D) ou la reconnaissance de formes en passant par les techniques de décision (M2 Rech. SETI, M2 Pro GSE).

Investissement dans les enseignements en Génie Informatique :

Les besoins en Génie Informatique étant très importants dans les filières d'enseignement dont je dépends, j'ai accepté de m'investir dans le montage et la responsabilité de nouveaux cours, TD et TP. En particulier le module "Conception orientée objet" du M1 IST. Par ailleurs, j'ai remplacé de manière ponctuelle des collègues partis en détachement, en délégation ou en congé dans des enseignements de Génie Informatique lorsque cela a été nécessaire (API Windows, UNIX, Génie Informatique, Informatique Générale).

Encadrement systématique de Travaux d'Etude et de Recherche :

Depuis mon recrutement, je me suis résolument investie dans l'encadrement des TER (Travaux d'Etude et de Recherche) du M1 IST ou TEI (Travaux d'Etude en Informatique) du L3 IST à raison de 3 à 6 étudiants par an en moyenne. En 2010, j'ai pris la responsabilité de l'organisation des TER pour la filière M1 IST.

Investissement dans de nouveaux domaines et filières :

En 2009, j'ai choisi de m'investir dans une tout autre formation : le Master Ergonomie de l'Université Paris Sud-XI. Dans cette filière, un module d'éclairagisme permet aux futurs ergonomes d'appréhender les problèmes de photométrie et de colorimétrie nécessaires à la bonne évaluation des ambiances d'éclairages. Ce cours a été un véritable défi pour ce qui me concerne en raison de l'hétérogénéité de la formation initiale des étudiantes (biologistes, psychologues, physiciens, etc.). Par ailleurs, les notions de physiques nécessaires au décryptage des normes d'éclairagisme doivent être explicitées et maîtrisées par certains étudiants dont le niveau dans cette discipline est très faible. Malgré la difficulté, cet enseignement a été très enrichissant en raison notamment des Travaux Pratiques. Les expériences de physiques réalisées pour caractériser des luminaires, estimer des éclairements ou des luminances, calculer des températures de couleurs, etc. m'ont permis de me rapprocher et de manipuler davantage certaines notions de physique non sans intérêt lorsque l'on s'intéresse à la vision artificielle. Mon intervention au Master Ergonomie a été associée à une intervention plus modeste dans la cadre de la formation permanente (DU Optométrie Spécialisée) et le Master 1 Optométrie Spécialisée de l'Université Paris Sud-XI.

En résumé, l'ensemble des disciplines enseignées correspondent bien à ma conception de mon domaine de recherche, qui peut être appréhendé comme un carrefour entre trois disciplines principales : le *Traitement des Images et la Vision*, *le Génie Informatique* (Langages C/Conception orientée objet/Systèmes Unix et windows) et enfin, la *Physique Appliquée* (éclairagisme/colorimétrie).

◦ **Allégements de services obtenus :**

- En 1999 et en 2000, suite à mon recrutement, j'ai bénéficié, comme tous les nouveaux recrutés au département de physique, d'un allégement de service à 150h Eq. TD.
- De 2002 à 2004, j'ai bénéficié de 10h eq. TD de réduction sur mon service d'enseignement en raison de mon investissement dans l'organisation des emplois du temps de Travaux Pratiques du DESS Génie des Systèmes Electroniques.
- En 2004 et en 2006, en raison de mes congés maternité, j'ai bénéficié d'un demi service d'enseignement.
- En 2007, j'ai pu bénéficier d'un demi-CRCT pour démarrer une nouvelle collaboration avec le LIVIC. Mon service a été, de ce fait, réduit de moitié.
- En 2010, j'ai bénéficié de 10h eq. TD de réduction sur mon service d'enseignement pour la gestion des Travaux d'Etudes et de Recherche du M1 IST.

◦ **Interventions dans les filières suivantes :**

L3 IST : L3 Information Système Technologie, Univ. Paris Sud-XI.
M1 Ergo. : M1 Ergonomie, Univ. Paris Sud-XI.
M1 Opto. : M1 Sciences de la vision, Optométrie spécialisée, Univ. Paris Sud-XI.
DU Opto. : DU Optométrie spécialisée, Univ. Paris Sud-XI.
M2P GSE : M2 Professionnel Génie des Systèmes Electroniques, Univ. Paris Sud-XI.
M1-IST : M1 Information Systèmes et Technologie, Univ. Paris Sud-XI.
M2R SETI : M2 Recherche Systèmes Electroniques et Traitement de l'Information, Univ. Paris Sud-XI.
ESME : 3ème année élèves Ingénieurs ESME-Sudria (Ecole Spéciale de Mécanique et d'Electricité), Ivry-Sur-Seine.
ESIGETEL : 3ème année Elèves Ingénieurs ESIGETEL (Ecole sup. d'ingénieurs en informatique et génie des télécom.), Avon.
DEUG SM/S4 : DEUG Sciences de la Matière, module "Robotique", Univ. Paris Sud-XI.
FIUPSO/IFIPS-3 : 3ème année Polytech. Paris Sud, Univ. Paris Sud-XI.
FIUPSO/IFIPS-2 : 2ème année Polytech. Paris Sud, Univ. Paris Sud-XI.

Synthèse : cours magistraux [5]

Filière	Intitulé du module	Heures	Année	Resp. du module
L3-IST	Génie Informatique	14	2010	*
M1 Ergo.	Eclairagisme et colorimétrie	24	depuis 2009	*
M1 Opto.	Vision et Eclairagisme	6	depuis 2009	*
DU Opto.	Vision et Eclairagisme	3	depuis 2009	*
L3-IST	Optimisation	1,5	2005-2008	
M2P GSE	Traitement d'images avancé	6	depuis 2003	*
M1 IST	Conception Orientée Objet	21	depuis 2002	*
M1 IST	Traitement d'images, signal	10	2007, 2009	
M2R SETI	Vision par ordinateur	6	2007, 2009	
M1 IST	Traitement d'images, Vision	18	2007	*
ESME	Vision industrielle	12	1995-2006	*
ESIGETEL	Vision par ordinateur	16	2002-2004	*

Synthèse : travaux dirigés [6]

Filière	Intitulé du module	Heures	Année
M1 IST	Conception Orientée Objet	3 à 5	depuis 2002
DEUG SM/S4	Reconnaissance de formes	30	2000-2002
M1 IST	Traitement d'images, signal	28	1999-2004

Synthèse : travaux Pratiques

Filière	Intitulé du module	Heures	Année	Montage
L3-IST	Génie Informatique	32	2010	
M1 Ergo	Eclairagisme et colorimétrie	8	depuis 2009	
L3-IST	Optimisation	3	depuis 2005	*
M1 IST	Conception Orientée Objet	24	depuis 2002	*
M2P GSE	Traitement d'images	24	depuis 1999	*
IFIPS-3	Traitement d'images	24	1999-2008	*
M2R SETI	Traitement d'images	16	depuis 1999	*
M2P GSE	Synthèse d'images	16	1999-2005	*
FIUPSO3	Synthèse d'images	16	1999-2005	*
M1 IST	Traitement d'images, signal	32	1999-2004	*
FIUPSO-2	Unix + API Windows	20	2001	
L3-IST	Génie Informatique	28	2001-2003	
ESIGETEL	Vision par ordinateur	8	2002-2004	*

5. Des polycopiés ont été rédigés pour tous les cours ci-dessous.
6. Un recueil d'exercice a systématiquement été rédigé pour les besoins des TD.

1.5 Activités administratives

Responsabilités administratives liées à l'enseignement

- Mise en place de nouveaux travaux pratiques : **développement de bibliothèques d'algorithmes de vision par ordinateur pour l'enseignement**. *Les algorithmes développés en C étaient d'abord intégrés dans les environnements Khoros, puis Visiquest. Actuellement, nous réalisons leur transcription en java pour une intégration dans ImageJ.*

- **Mise en place de plusieurs nouveaux modules** : Traitement et synthèse d'images, Traitement des images et du signal, Traitement des images et Vision, Conception orientée objet. *Rédaction des polycopiés associés, des sujets de travaux pratiques et travaux dirigés. Encadrement des différentes intervenants de TP (ATER, moniteurs).*

- **Responsabilités de 7 modules d'enseignement** en L3, M1, M2 Pro.

- **Participation active à la vie enseignante** : participation aux jurys, aux réunions pédagogiques, aux réunions de mise en place de nouvelles filières ou nouveaux modules.

- 2009/2010 : **Chargée de l'affectation et de l'organisation des TER** (travaux d'étude et de recherche) du M1 IST, Université Paris Sud-XI. *Collecte des sujets auprès des enseignants, affectation aux étudiants en fonction des choix, suivi et interface entre encadrants et étudiants, organisation des soutenances, participation aux jurys.*

- 2000-2009 : **Chargée du suivi de stages des étudiants** du M2 Pro Génie des Systèmes Electroniques (Université Paris Sud-XI) en entreprise (3 par an). *Visite en entreprise durant les stages, organisation de réunions avec les tuteurs en entreprise, organisation des soutenances, lecture des rapports, participation aux jurys.*

- 2002-2004 : **Chargée de l'organisation des emplois du temps des travaux pratiques** du DESS Electronique, Université Paris Sud-XI.

- 1999-2004 : **Responsable de la salle TP de Traitement d'image** de l'UFR ainsi que celle de la FIUPSO. Gestion informatique quotidienne des machines en réseaux sous Linux.

- 1999-2004 : **Participation aux entretiens de recrutement** pour l'entrée en FIUPSO (Polytech Paris Sud) et au M2 Pro Génie des Systèmes Electronique, Université Paris Sud-XI.

Responsabilités administratives liées à la recherche

- 2011 : **Délégué** de liste pour le collège B, section 61 au CNU. Le résultat des élections sont diffusés le 25 octobre 2011.
- Depuis 2009 : **Expert collège B du comité de sélection** sections 61-63 de l'Université Paris VI. *Participation aux commissions de recrutement pour les postes 955 (Systèmes robotiques, applications pour la santé), 1568 (Commande des systèmes robotiques) et 0363 (Perception pour la Robotique) en 2009, 2010 et 2011.*

- 2006-2008 : **Implication dans la vie de l'équipe** VISA . A titre d'exemple :

  - **Chargée de la présentation de l'équipe** de recherche VISA auprès des instances de l'AERES, exposé des activités de recherche de l'équipe en déc. 2008.
  - **Chargée de la présentation des partenariats industriels** de l'équipe VISA auprès des instances du CRITT (Centre Régional d'Innovation et de Transfert de Technologie) en 2007.

- 2003-2007 : **Membre élu (collège B) de la commission des spécialistes** 61$^{\text{ème}}$ section, Université Paris Sud-XI.

- 2002-2006 : **Membre élu (collège B) du conseil de laboratoire** de l'Institut d'Electronique Fondamentale (Université Paris Sud-XI).

1.6 Activités liées à la recherche

Rayonnement

- **Membre du comité de programme** de "International Conference On Neural Computation", session Pattern recognition en 2010 (Espagne) et en 2011 (France).

- **Membre du comité de programme** de IEEE "International Conference on Machine and Web Intelligence", Algiers, 3-5 oct, 2010.

- **Reviewer régulier** pour les revues "Traitement du Signal", "IET Computer Vision" et les conférences "IROS" (IEEE/RSJ International Conference on Intelligent RObots and Systems), ITSC (IEEE Conference on Intelligent Transportation systems).

- **Séminaire invité** : "Traitement d'images pour les systèmes embarqués". Ecole Militaire Polytechnique, Alger 15 mars 2007.

Participation significative à des projets

Depuis sept. 2010 : Participation au Projet Européen "SPY" (Surveillance imProved System). *Ce projet a pour but de réaliser de nouveaux systèmes d'assistance et de surveillance, intelligents et automatisés, adaptés pour la mobilité des utilisateurs*

(forces de sécurité) sur le terrain. Il s'inscrit dans le programme européen ITEA2 et a obtenu son label le 9 décembre 2009.
- Participation personnelle dans la partie "surveillance automatique par des approches monoculaires". Mon travail porte sur la compensation du mouvement du capteur et l'étiquetage des objets mobiles en fonction de la nature de leur mouvement.
- Encadrement d'un post-doc à compter de septembre 2011.

2008-2011 : Responsable du projet "STEREO" retenu lors de l'appel à projets 2008 conjoint Digiteo/Région Île-de-France pour le DIM Logiciels et systèmes complexes. *Ce projet concerne la coopération mouvement/stéréovision pour la détection d'obstacles à partir de caméras embarquées sur véhicule mobile. Les deux partenaires du projets sont l'IEF (Samia Bouchafa) et le LIVIC (Didier Aubert).* J'ai été moteur dans la création, la soumission, la recherche de partenaires et de doctorants dans ce projet.
- Co-encadrement d'un doctorant (Adrien Bak).
- Réalisation d'un système de détection d'obstacles à partir de caméras embarquées, basé sur de la coopération mouvement/stéréo.

2007-2008 : Participation au Projet "Love" (Logiciel d'Observation des VulnérablEs) labellisé par le pôle de compétitivité System@tic. *Ce projet propose de contribuer à la sécurité routière en mettant principalement l'accent sur la sécurité des piétons.*
- Contribution personnelle dans la partie SP2 concernant la détection d'obstacles par traitement d'images en particulier dans la partie "approches monoculaires pour l'analyse du mouvement". Encadrement d'un ingénieur en CDD sur le projet (Antoine Patri), développement d'algorithmes d'estimation du mouvement et de détection du FOE (Foyer d'Expansion).

1996-1998 : Participation au Projet "CROMATICA" (CROwd Management using Telematic Imaging and Communication Assistance, DG XIII, 4th PCRD). *Ce projet concerne la détection de comportements de foules anormaux dans le métro par traitement d'images (contresens, mouvements de panique, chutes dans les voies, etc.).*
- Dans le cadre de ce projet, qui constitua le cadre applicatif de ma thèse, j'ai été amenée à réaliser un système de détection de contresens dans les couloirs du métro. Je me suis impliquée dans l'élaboration des enquêtes destinées aux opérateurs de surveillance de la RATP, j'ai effectué la collecte des enregistrements (une demi journée par semaine pendant 6 mois). Le système mis en oeuvre a été installé notamment dans la station de métro Parisienne "Havre Caumartin".

Partenariats industriels et académiques

Depuis janv. 2011 : Partenariat récent avec le LIMSI, équipe AMI, dépôt d'un projet ANR en commun prévu en 2012. Je suis responsable avec Christian Jacquemin d'une AI (Action Incitative) financée par le LIMSI et l'IEF (acceptée en mars 2010,

durée : deux ans). *L'objectif de ces projets est de proposer une chaîne complète de développement d'application de réalité augmentée spécialisée (projection sur une surface quelconque d'éléments virtuels afin de l'enrichir).*

- Contribution personnelle sur la mise en correspondance d'images pour le recalage, la détection de surface planes (bâtiments/monuments architecturaux) et la compensation géométrique et photométrique sur ces surfaces.

Depuis oct. 2008 : Partenariat avec le LIVIC au travers du projet Digiteo "STEREO" (Appel à projet DIM systèmes complexes), voir section 1.6.

- Co-encadrement d'un doctorant.
- 3 publications ont été effectuées dans le cadre de cette collaboration.

Mai 2007 - Mars 2008 : Contrat de prestation de service avec la société PMC (Périphériques Matériels et Contrôle).

- J'ai réalisé dans le cadre cette prestation, un système de "Détection d'unicité de présence dans les locaux de valeur des agences bancaires".
- J'ai participé à l'encadrement, à plein temps pendant 3 mois, d'un ingénieur de développement en Traitement d'images de la société PMC : David Bernuau.
- Suite à ce partenariat, la société PMC a choisi l'Université Paris Sud-XI pour le versement annuel de sa taxe d'apprentissage (5000 à 7000 euros/an).
- Une nouveau projet est en cours de définition avec la société PMC.

Sept. 2003 : Contrat de prestation de service avec le LIVIC.

- Suite aux nouvelles orientations thématiques définies au sein du Livic, j'ai été chargée de réaliser une étude bibliographique sur les méthodes monoculaires d'analyse du mouvement 3D.
- J'ai rédigé un rapport au terme de ce contrat.

2000-2003 : Collaboration avec le département Minasys de l'IEF sur le recalage d'images microscopique.

- Co-encadrement de 2 stagiaires de Master Recherche avec Alain Bosseboeuf : Adel Hafiane et Najat Chihab.
- Une publication a été effectuée dans le cadre de cette collaboration.

1999-2002 : Collaboration avec l'Université de Louisiane sur les parcours récursifs d'images, en particulier avec le Prof. Guna Seetharaman qui a effectué plusieurs séjours invités à l'IEF dans le cadre de cette coopération.

- 2 publications ont été réalisées dans le cadre de cette collaboration.

1.7 Encadrement

Thèses soutenues	**2**	50%, 30%
Thèses en cours	**3**	50%, 50%, 50%
Stages de Master Recherche	**6**	
Stages de Master Professionnel	**3**	
Magistères ou projets de fin d'études	**5**	

Thèses soutenues

2. **Nikom Suvonvorn**. Co-encadrement à 50%. Directeur de thèse : Bertrand Zavidovique, IEF, Université Paris Sud-XI. Titre : "Mise en correspondance d'images pour l'analyse du mouvement et la stéréovision".

   - Bourse de coopération franco-thaïlandaise.
   - Thèse soutenue le 18 décembre 2006.
   - 3 publications effectuées avec le doctorant.
   - Devenir du doctorant : enseignant chercheur à Faculty of Engineering, Prince of Songkla University (Hatyai, Thaïlande).

1. **Fang Zi**. Co-encadrement à 30%. Directeurs de thèse : Bertrand Zavidovique, IEF, Université Paris Sud-XI et LI Yuanjun, Northwestern Polytechnical University (Xian, Shaanxi, Chine). Titre : "Contrôle visuel d'une voiture autonome et fusion multi-capteurs basée sur le traitement d'images multi-bandes".

   - Bourse en co-tutelle avec la Chine.
   - Soutenue le 25 novembre 2010.
   - Devenir du doctorant : secrétaire générale à Northwestern Polytechnical University (Xian, Shaanxi, Chine)

Thèses en cours

3. **Yasser Almehio**. Depuis janv. 2008. Co-encadrement à 50%. Directeur de thèse : Bertrand Zavidovique, IEF, Université Paris Sud-XI. Titre : "Profondeur, couleur et mouvement pour la détection et le suivi d'objets".

   - Bourse de coopération franco-syrienne + soutien local sur fond propre VISA.
   - Soutenance prévue en décembre 2011.
   - 4 publications effectuées avec le doctorant, 1 soumise.
   - Devenir du doctorant à compter de la rentrée 2011/2012 : poste de demi ATER à l'Université Paris Sud-XI

2. **Adrien Bak**. Depuis oct. 2008. Co-encadrement à 50%. Directeur de thèse : Didier Aubert, LIVIC (INRETS/LCPC). Titre : "Coopération stéréo/mouvement pour la détection des objets dynamiques".

- Projet retenu dans le cadre d'un appel d'offre DIGITEO, financement de thèse DIGITEO.
- Date de soutenance fixée pour le 14 octobre 2011, manuscrit remis aux rapporteurs.
- 3 publication effectuées avec le doctorant.
- Devenir du doctorant : depuis le 1er juillet 2011, ingénieur en CDI à DXO Labs.

1. **Qiong Nie**. Depuis sept. 2010. Co-encadrement à 50%. Directeur de thèse : Alain Mérigot, Université Paris Sud-XI. Titre : "Architectures pour algorithmes d'estimation du mouvement".

   - Financement de thèse MENRT.
   - Soutenance prévue en septembre 2013.

Stages de Master 2 Recherche

6. Cong Vinh. *Stéréovision par mise en correspondance de lignes de niveaux*, Master 2 Recherche SETI, Université Paris Sud-XI, 2005.
5. Fadi Abdin. *Groupement de lignes pour le recalage d'images*, Master 2 Recherche SETI, Université Paris Sud-XI, 2005.
4. Arezki Ait Seddik. *Stéréovision monoculaire*, Master 2 Recherche SETI, Université Paris Sud-XI, 2003.
3. Adel Hafiane. *Traitement d'images pour la caractérisation des vibrations des MEMS avec une résolution sub-pixel*, Master 2 Recherche SETI, Université Paris Sud-XI, 2002. **1 publication effectuée avec le stagiaire à l'issue du stage**.
2. Mohamed Trabelsi. *Localisation et reconnaissance de plaques d'immatriculation*, Master 2 Recherche SETI, Université Paris Sud-XI, 2002.
1. Najat Chihab. *Raccord de profils 3D issus de la microscopie interférométrique à balayage de franges*, Master 2 Recherche SETI, Université Paris Sud-XI, 2001.

Stages de Master 2 Professionnel

3. Mohamed Elhabouz. *Analyse d'images pour la vidéo-surveillance dans les banques*, M2 Pro. Informatique industrielle, Université Paris Sud-XI, 2006.
2. Charazade Azzoug. *Exploitation de la parallaxe pour l'estimation du mouvement 3D d'une caméra embarquée*, M2 Pro. Génie des Systèmes Electroniques, Université Paris Sud-XI, 2004.
1. Sylvain Marsault. *Estimation du mouvement et suivi de lignes blanches à partir d'une caméra embarquée sur véhicule*, M2 Pro. Génie des Systèmes Electroniques, Université Paris Sud-XI, 2002

Magistères ou projets de fin d'études

5. Cheng Chen et Yunxia Yao. *Mise en place d'un environnement de travaux pratiques pour les TP de traitement d'images sous ImageJ*, stage de Bachelor, niveau L3, Université Paris Sud-XI dans le cadre du programme franco-chinois d'accueil d'étudiants de l'Université de Huazhong (HUST), 2010.
4. Hichame Ouzaarou. *Détection de plans 3D par accumulation des vecteurs vitesse*, stage Magistère ENS-Cachan, 2009.
3. Thibaud Sénéchal. *Détection du mouvement par image de référence*, stage Magistère ENS-Cachan, 2006.
2. Thomas Kunlin. *Compensation de mouvement basée sur des directions locales de lignes de niveaux*, projet de fin d'études, ENSI de Strasbourg, 2001.
1. Bastien Jacquot. *Mise en correspondance de primitives basées sur les lignes de niveaux*, stage Magistère ENS-Cachan, 2001.

1.8 Liste de publications

Revues internationales avec comité de lecture	**8**
Revues nationales avec comité de lecture	**1**
Chapitres de livres	**3**
Conférences internationales avec actes	**24**
Conférences nationales avec actes	**2**
Conférences invitées dans des congrès internationaux	**3**
Rapports de recherche	**6**
Séminaires invités	**4**

Articles dans des revues internationales avec comité de lecture

8. A. Bak, S. Bouchafa, D. Aubert. "Dynamic Objects Detection through Visual Odometry and Stereo-Vision : A Study of Unaccuracy and Improvement Sources". **Machine Vision and Applications**, Special Issue on Car Navigation and Vehicle Systems, Springer. Acceptée avec révisions mineures. ISSN 0932-8092.
7. S. Bouchafa, B. Zavidovique. "C-velocity : a flow-cumulating uncalibrated approach for 3D plane detection". **International Journal of Computer Vision**, Springer, 2011. DOI : 10.1007/s11263-011-0475-6. ISSN 0920-5691.
6. S. Bouchafa, B. Zavidovique. "Error sources and their influence on C-velocity methods". **Pattern recognition and Image Analysis**, vol. 21(3), Pleiades Publishing, Ltd., Springer, 2011. *A paraître.* ISSN 1054-6618.
5. S. Bouchafa, B. Zavidovique. "Robustness of C-velocity methods for 3D plane detection" [7]. **Pattern recognition and Image Analysis**, vol. 21(2), Pleiades Publishing, Ltd., Springer, 2011, pp. 233-237, DOI : 10.1134/S1054661811020179. ISSN 1054-6618.

7. Numéro spécial regroupant les articles de la conférence PRIA 2010

4. S. Bouchafa, B. Zavidovique. "C-velocity : a Cumulative Frame to Segment Objects from Egomotion". **Pattern recognition and Image Analysis**, vol. 19(4), pp. 583-590. Pleiades Publishing, Ltd., Springer, 2009. ISSN 1054-6618.
3. S. Bouchafa, B. Zavidovique. "Efficient cumulative matching for image registration". **Image and Vision Computing**, Elsevier, vol. 24(1), pp. 70-79, 2006. ISSN 0262-8856.
2. D. Aubert, F. Guichard, S. Bouchafa. "Time-scale change detection applied to real time abnormal stationarrity monitoring". **Real Time Imaging**[8], vol. 10, Issue 1, pp. 9-22, 2004. ISSN 1077-2014.
1. S. Bouchafa, D. Aubert, L. Beheim, A. Sadji. "Automatic counterflow detection in subway corridors by image processing". **Intelligent Transportation Systems Journal**[9], vol. 6, Issue 2, pp. 97-123, Overseas Publishers Association, 2001. ISSN 1024-8072.

Articles dans des revues nationales avec comité de lecture

1. D. Aubert, S. Bouchafa. "Détection de stationnarités anormales dans les couloirs du métro". **Revue Transport et Sécurité**[10], n°62, pp. 56-72, Elsevier, 1999. ISSN 0761-8980.

Chapitres d'ouvrages collectifs

3. A. Bak, S. Bouchafa, D. Aubert. "Focus of Expansion Localization Through Inverse C-velocity"[11]. **Image Processing, Computer Vision Pattern Recognition and Graphics, SL6, Lecture Notes in Computer Science** 6978, Springer Berlin 2011, pp. 484-493, ISBN 978-3-642-24084-3.
2. N. Suvonvorn, S. Bouchafa, B. Zavidovique. "Marrying Level Lines for Stereo or Motion"[12]. **Image Analysis and Recognition, Lecture Notes in Computer Science**, Springer Berlin / Heidelberg, pp. 391-398, vol. 3656, 2005. ISSN 0302-9743.
1. F. Guichard, S. Bouchafa, D. Aubert. "A Change Detector Based on Level Sets". **Mathematical Morphology and its Applications to Image and Signal Processing, Computational Imaging and Vision**, Springer US, Kluwer academic Publishers, pp. 321-330, vol. 18, 2002. ISBN 0-792-37862-8.

Conférences internationales avec actes et comité de sélection

24. A. Bak, S. Bouchafa, D. Aubert. "Focus of Expansion Localization Through Inverse C-velocity". **International Conference on Image Analysis and Processing** (ICIAP 2011), 14-16 sept, Ravenna, Italia.

8. Depuis 2006, Real-time Imaging n'est plus édité. Certains éditeurs de la revue ont crée depuis Real-time Image Processing, Springer.
9. Depuis 2002, ITS journal devient : Journal of Intelligent Transportation Systems ISSN : 1547-2450
10. Depuis 2010 : RTS est édité par Springer.
11. Numéro spécial regroupant les articles de la conférence ICIAP 2011
12. Numéro spécial regroupant les articles de la conférence ICIAR 2005

23. S. Bouchafa, B. Zavidovique. "Robustness of C-velocity based methods for 3D moving plane detection ". **Pattern Recognition and Image Analysis** (PRIA), pp. 177–181, St Petersburg, Russia, dec., 2010.
22. A. Bak, S. Bouchafa, D. Aubert. "Detection of independent-moving objects through stereo-vision and ego-motion extract". **IEEE Intelligent Vehicules Symposium** (IV), pp. 863–870, San Diego CA, june 21-24, 2010.
21. Y. Almehio, S. Bouchafa. "Matching images using invariant level-line primitives under projective transformation". Seventh **Canadian Conference on Computer and Robot Vision** (CRV), IEEE computer society, pp. 130–135, , Otawa, Canada, may 31-june 2, 2010.
20. Y. Almehio, S. Bouchafa. "Robust Projective Transformation Estimation Using Invariant Level-line Primitives". **IEEE southwest Symposium on Image Analysis and Interpretation** (SSIAI), pp. 57–60, 23-25 may 2010, Austin, USA.
19. Y. Almehio, S. Bouchafa. "A voting decision strategy for image registration under affine transformation". Image Processing : Algorithms and Systems VIII" conference, **SPIE Electronic Imaging**, vol. 7532, pp. 75320B, San Jose, California, USA, jan. 17-21, 2010.
18. S. Bouchafa, A. Patri, B. Zavidovique. "Efficient plane detection from a single moving camera". **IEEE International Conference on Image Processing** (ICIP), pp. 3457–3460, Cairo, Egypt, nov. 7-11, 2009.
17. S. Bouchafa, A. Patri, B. Zavidovique. "Exhibiting planar structures from egomotion". **International Conference on Informatics in Control, Automation ans Robotics** (ICINCO), pp. 183–188, Milan, Italy, 2-5 july 2009.
16. M. Gouiffès, S. Bouchafa, B. Zavidovique. "Segments of color lines - A Comparison through a Tracking Procedure". **International Conference on Informatics in Control, Automation ans Robotics** (ICINCO), pp. 433-438, Milan, Italy, 2-5 july 2009.
15. Y. Almehio, S. Bouchafa. "Efficient Affine Motion Estimation Using Level-Lines Grouping". **International Conference on Image Processing, Computer Vision, and Pattern Recognition** (IPCV), pp. 807–813, Las Vegas, USA, july 13-16, 2009.
14. S. Bouchafa, B. Zavidovique. "Moving plane detection under translationnal camera motion using the C-Velocity concept". **IEEE Inter. Workshop on Image Processing, Theory tools and Application** (IPTA), pp 1-8, Sousse, Tunisia, 24-27 nov., 2008.
13. S. Bouchafa, B. Zavidovique. "C-Velocity : cumulative identification of moving planes". 9th Conf. on **Pattern Recognition and Image Analysis** (PRIA), pp. 59–62, Nizhny Novgorod, Russia, sept. 15-21, 2008.
12. N. Suvonvorn, S. Bouchafa and B. Zavidovique. "Marrying level lines for stereo or motion". **International Conference on Image Analysis and Recognition** (ICIAR), pp. 391–398, Toronto, Canana, sept. 28-30, 2005.
11. S. Bouchafa, B. Zavidovique. "A voting strategy for level-line matching". **IEEE Advanced Concepts for Intelligent Vision Systems** (ACIVS) aug. 31- sept 3, Brussels, Belgium, 2004.

10. N. Suvonvorn, S. Bouchafa, L. Lacassagne. "Fast Reliable Level-Lines Segments Extraction". **IEEE international conference on Information Communication Technologies : from Theory to Applications**, pp. pp. 349–350, Damascus, Syria, april 19-23, 2004.
9. A. Hafiane, S. Petitgrand, Olivier Gigan, S. Bouchafa, A. Bosseboeuf. "Study of subpixel image processing algorithms for MEMS in-plane vibration measurements by stroboscopic microscopy". **SPIE Microsystems Engineering : Metrology and Inspection III**, vol. 5145, 169–180, Munich, june 2003.
8. S. Bouchafa, B. Zavidovique."Cumulative level-line matching for image registration". **IEEE International Conference on Image Analysis and Processing** (ICIAP), pp. 176–181, sept. 17-19, 2003.
7. S. Bouchafa, B, Zavidovique. "Efficient line voting for MEMS profile registration". **Indian Conference on Computer Vision Graphics and Image Processing** (ICVGIP), Ahmedabab, India, dec., 2002.
6. G. Seetharaman, S. Bouchafa, B. Zavidovique. "Concurrent edge/region detection using a Peano scan". 11^{th} **International Conference On Image Analysis and Processing** (ICIAP), pp. 125–130, Palerme, sept., 2001.
5. S. Bouchafa, G. Seetharaman, B. Zavidovique. "Large image decimation and reconstruction using a Peano scanning". IASTED **International Conference on Computer Graphics and Imaging** (CGIM), Hawai, aug., 2001.
4. F. Guichard, S. Bouchafa, D. Aubert. "A change detector based on level sets". **International Symposium on Mathematical Morphology** (ISMM), pp. 322–330, Palo Alto, juin 2000.
3. L. Khoudour, JP Deparis, JL Bruyelle, F. Cabestaing, D. Aubert, S. Bouchafa, S. Velastin. "The CROMATICA project". **IEEE international Conference on Image Analysis and Processing** (ICIAP), pp. 757–764, Florence, sept. 1997.
2. S. Bouchafa, L. Beheim, A. Sadji, D. Aubert. "Crowd motion estimation in subway corridors using image processing". 4th world **congress on Intelligent Transport Systems** (ITS), Berlin , oct. 1997.
1. S. Bouchafa, D. Aubert, S. Bouzar. "Crowd motion estimation and motionless detection in subway corridors by image processing". **IEEE Intelligent Transport Systems Conference** (ITSC), pp. 332–337, Boston, nov., 1997.

Conférences nationales avec actes et comité de sélection

2. S. Bouchafa, B. Zavidovique. "Stratégie de vote pour la mise en correspondance de lignes de niveaux". **Congrès Reconnaissance de Formes et Intelligence Artificielle** (RFIA), toulouse, volume 2, pp. 605-614, janv. 2004.
1. D. Aubert, S. Bouchafa. "Détection de stationnarités anormales dans les couloirs du métro". **Congrès International Francophone, ATEC**, janv., 1999.

Conférences invitées dans des congrès internationaux

3. S. Bouchafa. "Analyse du mouvement 2D/3D ". Tutorial session "Analyse d'images", International Symposium on Programming and Systems, Alger, 9-11 mai 2005.
2. S. Bouchafa, V. Di Gesu, C. Valenti, B. Zavidovique. "Symmetry based operators and their application in computer vision and pattern analysis". Workshop Distributed Processing, Transfer, Retrieval, Fusion and Display of Images and Signals : High Resolution and Low Resolution in Data and Information Grids, Granada, Spain, 21-22 February, 2003.
1. S. Bouchafa. "Automatic observation of abnormal behaviours". European Conference on Transport Psychology, Angers, juin 1999.

Rapports de recherche, séminaires invités

Rapports de recherche

6. S. Bouchafa. "Approches monoculaires pour l'estimation du mouvement 3D". Contrat de prestation de service IEF/LIVIC. Sept. 2003.
5. S. Bouchafa. "Détection du mouvement insensible aux variations du contraste. Application a la détection de comportements anormaux dans le métro par traitement d'images". Thèse de doctorat, Université Paris VI, décembre 1998.
4. J-M. Blosseville, S. Bouchafa, M. Cottinet, MH. Massot, A. Polacchini. "Evaluation technico-économique d'un système de voitures en libre service : le système PRAXITELE". Rapport INRETS Arcueil, France, 1996.
3. S. Bouchafa. "Les transformations d'images en général, la transformation de Hadamard en particulier". Rapport interne. Equipe vidéo, INRETS, juin 1997.
2. S. Bouchafa. "Traqueur 3D par traitement d'image pour une application de la Réalité Virtuelle". Rapport de stage de DEA, Laboratoire de Robotique de Paris, 1994.
1. S. Bouchafa, H. Seghouani. "Appariement d'une séquence d'images prises par une caméra en rotation". Rapport de projet de fin d'études, Institut National d'Informatique, Alger, sept. 1993.

Séminaires invités

4. S. Bouchafa. "Traitement d'images pour les systèmes embarqués". Ecole Militaire Polytechnique, Alger, 15 mars 2007.
3. S. Bouchafa. "Détection de mouvement anormaux dans le métro par traitement d'images", Groupe de travail amélioration de la qualité de l'offre des transports collectifs (INRETS/RATP), 1999.
2. S. Bouchafa. "Méthodes d'analyse du mouvement". Séminaire du DEA Robotique, Université Paris VI, 1998.
1. S. Bouchafa. "Détection du mouvement par lignes de niveaux". Séminaire du DESS Imagerie Electronique, Université Paris VI, 1998.

Chapitre 2

Résumé des travaux de recherche, ordre chronologique

2.1 Parcours de recherche

Mon intérêt pour le traitement d'images et la vision par ordinateur est apparu dès ma formation d'ingénieur à l'issue de laquelle j'ai choisi un **projet de fin d'études** portant sur la mise en correspondance d'images issues d'une caméra en mouvement. Le point de vue adopté durant cette étude était purement géométrique : il s'agissait de mettre en place un algorithme de prédiction/vérification des hypothèses de mise en correspondance, connaissant le mouvement du capteur. Cette étude m'a permis d'appréhender définitivement tous les aspects géométriques du mouvement et de la stéréovision, aspects peu complexes mais parfois rendus fastidieux par la nature et la multiplication des hypothèses.

Durant mon stage de DEA, j'ai poursuivi l'étude de la vision 3D à travers la réalisation d'un système de positionnement 3D du porteur d'un casque de réalité virtuelle à l'aide d'une caméra fixée au plafond . Des motifs suffisamment discriminants pour être facilement extraits automatiquement à l'aide des images acquises sont placés sur le casque. La correspondance établie entre coordonnées 3D de ces motifs (par rapport à un repère choisi sur le casque) et coordonnées 2D sur l'image, permettait alors d'estimer les transformations à 6 degrés de liberté que la tête du porteur de casque avait subies. Cette approche s'affranchissait ainsi des traqueurs classiques - exploitant d'autres types de capteurs - plus intrusifs et encombrants qu'une caméra.

Durant ma thèse, le thème principal de ma recherche s'est construit autours de l'analyse du mouvement à partir d'une caméra fixe. Une des principales préoccupations des chercheurs dans ce domaine était d'extraire des images des primitives insensibles aux variations pouvant affecter les niveaux de gris entre images successives, en particulier les variations d'illumination. L'objectif initial de la thèse était d'élaborer une transformation de l'image permettant de construire des primitives géométriques locales s'affranchissant des niveaux de gris eux même et se basant uniquement sur les relations d'ordre entre niveaux

de gris. Je me suis donc d'abord intéressée au transformations d'images telle que la transformation d'Hadamard, cette piste n'a pas été fructueuse en raison du caractère global de la transformation, mais elle m'a néanmoins conduite vers la représentation par lignes de niveaux, qui a complètement répondu à mes attentes.

Mon travail de thèse comportait alors trois volets. **Dans le premier volet**, une nouvelle méthode de détection du mouvement insensible aux variations d'illumination est proposée en partant de l'hypothèse que la caméra est fixe. Notre démarche s'appuyait sur une constatation triviale mais néanmoins importante : il est impossible, si l'on établit pour la détection du mouvement des critères basés exclusivement sur les variations d'intensité lumineuse, de distinguer entre la présence de mouvement dans la scène et les changements de contraste pouvant survenir. Nous avons alors montré que la représentation par ensemble de niveaux fournit une réponse satisfaisante en raison de sa robustesse. Ainsi, les effets distincts des changements d'illumination et du mouvement sur les lignes de niveaux sont indépendamment mis en évidence. Deux conséquences directes de la présence de mouvements ont été exploitées : l'apparition de nouvelles lignes de niveaux et les croisements de lignes de niveaux entre une référence et l'image courante. A partir de celles-ci, ont été établis les critères de détection. Des mesures locales sur les lignes de niveaux sont alors extraites de l'image grâce à un processus de suivi conduisant ainsi à une caractérisation par orientation locale. **Dans le second volet**, les détections obtenues sont exploitées dans le but d'assurer l'estimation du mouvement. Des variantes de méthodes existantes, à savoir la mise en correspondance de blocs et la méthode de Horn et Schunk, sont alors proposées. Enfin, **le dernier volet de la thèse** répond à son cadre applicatif et correspond à un besoin exprimé par certains réseaux de transports en commun : la détection de comportements anormaux dans les couloirs du métro, plus particulièrement les contresens et les stationnarités de trop longue durée. Les méthodes développées régissent sans difficulté la détection de ces deux types de situations.

Après ma thèse, en janvier 1999, j'ai été recrutée dans la société CITILOG (Carrefour Intelligent Traitement d'Images Logiciels) sur un poste d'ingénieur de développement en Traitement d'images. J'ai été chargée de travailler sur un projet en collaboration avec "Voies Navigables de France" dont le thème était la détection de présence sur les ponts mobiles. J'ai participé à la mise en oeuvre d'un algorithme de détection de piétons et véhicules expérimenté sur le pont de Zuydcoote (Pas de Calais).

A mon arrivée à l'IEF, en septembre 1999, j'ai d'abord été amenée à travailler sur les thèmes traités par mon équipe d'accueil, en particulier : les parcours récursifs d'images. Après avoir défini puis étudié les balayages de Peano et des critères d'optimalité du type "variation de niveaux minimale (resp. maximale) le long du parcours", nous les avons appliqués à la recherche de redondances dans l'image puis à la compression par programmation dynamique dans le cadre d'une collaboration avec l'Université de Louisiane.

Dans les années qui ont suivi, nous avons successivement proposé :

- Une approche robuste de recalage cumulatif d'images dédié aux transformations géométriques simples.
- Un recalage d'images pour les transformations complexes.
- Une approche de mise en correspondance d'images pour l'analyse du mouvement et la stéréovision.
- Un approche de coopération mouvement/stéréovision pour la détection d'obstacles à partir de caméras embarquées.
- Enfin, une nouvelle méthode de détection d'obstacles basée sur un espace de représentation original, cumulatif, appelé *c-vélocité*.

Les applications n'ont pas manqué ; outre celles liées à la conduite automobile, nous avons étudié :

- Une application du recalage à la microscopie électronique.
- Un système de détection d'unicité de présence.

Ces thèmes, confirmant ainsi mon intérêt pour la vision dynamique des systèmes, seront développés ci-dessous.

2.2 Thèmes de recherche développés à l'IEF

2.2.1 Recalage cumulatif d'images

Alors que je travaillais sur les parcours récursifs d'images en collaboration avec l'Université de Louisiane, il m'est apparu indispensable de continuer, dans l'esprit de mon travail de thèse, à explorer les propriétés de robustesse des lignes de niveaux dans des applications où cette robustesse est recherchée. Mes activités de recherche dans ce secteur se sont dès lors focalisées sur la mise en correspondance d'images en général, étape indispensable en stéréovision ou en analyse du mouvement et où les approches classiques souffrent d'un manque de robustesse des primitives à apparier. L'hypothèse de départ est que les images considérées peuvent être acquises à partir de points de vues différents, à deux instants différents ou à partir de capteurs différents. Nous proposons alors une approche basée sur l'exploitation des lignes de niveaux, robustes vis-à-vis de variations de contraste. De plus, elles sont particulièrement adaptées à une stratégie d'appariement basée sur un processus de **vote**. En effet, le nombre important de lignes de niveaux dans l'image y entraîne de fortes redondances locales et permet d'envisager un processus de mise en correspondance **cumulatif** où chaque ligne est appelée au vote à une phase donnée en fonction de sa fiabilité. Pour restreindre l'espace de vote, nous exploitons une technique de décision sur graphe bi-partite dite des mariages stables. Chaque primitive crée une liste de préférence des primitives dans l'autre image, classées des plus aux moins ressemblantes en fonction de métriques adaptées. Ces préférences pondèrent le processus de décision se déroulant en plusieurs étapes : les primitives les plus fiables participent d'abord à un vote, les autres sont sollicitées ensuite pour confirmer/infirmer le vote précédent.

2.2.2 Application au recalage d'images microscopiques

Pour valider notre méthode de mise en correspondance, nous avons établi une collaboration avec le département Minasys (MIcro et NAnoSYStèmes) de l'IEF et accueilli un chercheur viétnamien (Ha Vu Le). Les chercheurs de ce département disposent d'un microscope électronique à balayage de franges dont le champs "visuel" est limité. Afin de disposer de la totalité de l'image du micro-dispositif à analyser, il est nécessaire de raccorder plusieurs prises de vues. Ce raccord, jusque là manuel, pourrait être réalisé de manière automatique à condition de savoir mettre en correspondance les images issues du microscope et estimer les transformations 2D entre ces images. La difficulté étant la présence de nombreux motifs répétitifs dans ces images (ex. peigne de nano capacités dans un accéléromètre type MEMS), ainsi que leur très faible contraste. Nous avons comparé notre approche avec trois techniques classiques : une corrélation de phases, une méthode d'optimisation (Levenberg-Marquardt) et une approche différentielle (démonstration web en ligne de University of California, Santa Barbara). Aucune des trois n'a donné les résultats escomptés. Notre approche basée sur la mise en correspondances cumulative de lignes de niveaux y parvient parfaitement, ce qui encourage à la généraliser à des transformations 3D plus complexes.

2.2.3 Mise en correspondance pour l'analyse du mouvement et la stéréovision

Le travail de thèse de **Nikom Suvonvorn** peut être considéré comme une recherche pragmatique de compromis entre l'effort de modélisation (i.e. qualification puis quantification) des objets, plus précisément des primitives objets, à mettre en correspondance et celui de mise en correspondance proprement dite. En effet, le but était de trouver une méthode efficace de résolution du problème de l'appariement d'images en général, ne préjugeant pas de l'application (reconnaissance, mouvement ou stéréovision). Nous avons donc été amenés à la fois à proposer des primitives que nous considérons particulièrement robustes, les jonctions de lignes de niveaux, et à retenir un paradigme d'optimisation que nous considérons particulièrement riche, les mariages stables. Un système générique pour la mise en correspondance d'images utilisant ces algorithmes a donc été élaboré. Le fonctionnement est conçu pour l'appariement en multi-échelles des jonctions de lignes de niveaux. L'efficacité définitive est cette fois testée par comparaison avec des méthodes de vote et de programmation dynamique déjà considérées très performantes dans leurs applications respectives : le recalage bidimensionnel et la détection d'obstacle par stéréovision.

Les résultats montrent une amélioration des performances importante : par exemple une voiture arrivant sur la file opposée est détectée par notre véhicule PICAR lui-même en mouvement à une cinquantaine de mètres en moyenne au lieu de la vingtaine obtenue jusque là avec un algorithme type Viterbi par lignes. Plus important encore, un cycliste (donc un piéton) sera détecté quasi sans interruption (90% du temps) par notre système à 36 mètres alors que sa taille dans l'image n'est que de 250 pixels, contre une détection de l'ordre de 10% par la version précédente du système visuel de PICAR. Les jonctions de lignes de niveau et les algorithmes de mariages stables sont réemployés dans notre système d'analyse du mouvement. Il est capable d'identifier des objets mobiles par leurs déplacements relatifs

puis de les extraire en étant lui-même en mouvement. Une méthode de classification basée sur les C-moyennes floues avec contrainte spatiale sur l'ensemble de niveaux a été définie.

La segmentation d'objet d'après leur mouvement est quant à elle fondée sur la méthode dite JF-Snake, adaptation d'une méthode de contours actifs conventionnelle au cas de nos jonctions de lignes de niveaux.

2.2.4 Recalage d'images avec prise en compte de transformations complexes

Dans la thèse de **Yasser Almehio**, nous tentons de généraliser notre approche de recalage d'images pour tenir compte de transformations plus complexes, ce qui implique la prise en compte d'invariants différents et d'espaces de vote de dimension plus importante. A partir des lignes de niveaux, sont construites des primitives adaptées aux invariants choisis, eux-mêmes dépendants des transformations recherchées. A titre d'exemple, pour une simple estimation de transformation 2D type "translation 2D", des segments de lignes de niveaux sont détectés.

Pour une transformations type "similarité", des angles (couples de segments) sont extraits. Pour une transformation affine, des Z et X sont construits. Enfin, pour une transformation projective complète, des quadruplets de segments sont associés pour calculer des bi-rapports. La gestion d'espace de vote de taille importante avec mise en place de stratégies de vote progressives est au coeur de cette thèse.

2.2.5 Détection d'unicité de présence

J'ai développé un algorithme de détection du mouvement basé sur l'extraction de primitives issues des lignes de niveaux, leur caractérisation et la comparaison des caractéristiques extraites entre images successives. Cet algorithme est une généralisation de l'algorithme de détection de mouvement proposé durant ma thèse. Il a été testé et comparé à des algorithmes classiques de détection sur des scènes où les variations de luminosité sont fréquentes.

Les résultats obtenus m'ont permis de réaliser un transfert industriel en mai 2007 à travers un contrat de prestation de service avec la société **PMC** (Périphérique Matériels et Contrôles) pour la réalisation d'un système de détection d'unicité de présence dans les locaux de valeur des agences bancaires à partir de caméras de vidéo-surveillance.

2.2.6 Coopération stéréovision/mouvement pour la détection d'obstacles

Dans le cadre du projet STEREO (appel à projet DIM systèmes complexes DIGITEO) et de la thèse d'**Adrien Bak**, nous nous intéressons à la détection d'obstacles à partir de caméras embarquées sur véhicules mobiles, visant la mise en place de systèmes d'aide à la conduite. L'objectif de ce travail est d'obtenir une méthode permettant une détection fine des objets dynamiques à l'aide d'un capteur de stéréo-vision, lui même en mouvement. Nous devons donc estimer le mouvement de notre système de caméra, avant de trouver un moyen permettant d'identifier les objets ayant leur propre mobilité.

L'extraction de l'ego-mouvement est réalisée à partir de deux cartes de disparité successives et des deux images issues d'un des capteurs. Un ensemble de points d'intérêt robustes (SURF) est extrait de chacune des deux images. A partir de cet ensemble de points, nous résolvons le système d'équations découlant de la projection du mouvement, grâce à un processus de RANSAC, insensible à la présence de faux appariements ou aux mouvements minoritaires. A aucun moment, nous ne faisons intervenir les coordonnées de nos points dans l'espace 3D. En effet, si le bruit dans l'espace image est entièrement isotrope, cela n'est pas vrai dans l'espace objet.

Cette méthode estime les six paramètres du mouvement de notre capteur stéréo entre deux poses successives, ce qui nous permet ensuite de détecter les objets dynamiques de la scène en confrontant chaque point de l'image au modèle de mouvement estimé.

2.2.7 *c-vélocité* : un nouvel espace de vote cumulatif pour la détection de surfaces paramétrées

Notre travail sur la mise en correspondance d'images nous a permis de confirmer la robustesse des techniques cumulatives rejoignant ainsi la théorie - en Intelligence Artificielle - des accumulations d'évidences. Par ailleurs, ma collaboration avec le LIVIC dans le cadre du projet Love, ainsi qu'au travers de l'encadrement commun d'Adrien Bak, m'a réellement confortée dans la recherche d'un cadre théorique plus large aux techniques de vote. Le LIVIC a en effet déjà proposé une approche de détection d'obstacles basée sur la stéréovision et la mise en place d'un espace de vote appelé *v-disparité* : les disparités étant cumulées le long des lignes image. Il m'a semblé très intéressant de montrer que la *v-disparité* n'était qu'un cas particulier d'un paradigme de vision plus général.

Pour cela, une première étape a été de la généraliser au mouvement 3D par la définition d'un espace de vote que nous avons appelé alors *c-vélocité* par analogie à la *v-disparité*. Les vitesses 2D apparentes sont cumulées le long de courbes d'iso-vitesse pour obtenir un espace de vote dans lequel des plans 3D par exemple sont représentés par des paraboles. Cette technique a été appliquée avec succès à la détection d'obstacles à partir d'une caméra embarquée dans le cas d'un déplacement du véhicule majoritairement translationnel. Elle ne nécessite aucune calibration préalable et aucune connaissance *a priori* du mouvement propre (ego-mouvement) de la caméra. Les premières expérimentations issues de cette approche sont très prometteuses et confirment ainsi la robustesse recherchée. En effet, malgré une mauvaise qualité d'estimation des vitesses 2D (un flot optique classique est employé), ajoutée à la présence de rotations d'amplitudes non négligeables, les résultats escomptés ont bien été atteints.

Deuxième partie

Synthèse des travaux de recherche

Chapitre 3

Introduction

C'est au sein de l'Institut d'Electronique Fondamentale que mes travaux de recherche ont été réalisés. L'équipe VISA (Anciennement **A**rchitectures **P**our la **V**ision), dirigée par le Prof. Bertrand Zavidovique, du département ACCIS m'a accueillie en 1999. Les activités de l'équipe s'étaient concentrées notamment sur l'élaboration d'architectures matérielles dédiées à la vision par ordinateur. Dans les rétines conçues et étudiés au sein de l'équipe, la nécessité de définir un ensemble d'objets et d'opérateurs en nombre limité et réutilisables a permis la définition d'une vision nommée "vision fruste" [Cou95] guidée par l'action, opportuniste, mettant en oeuvre les moyens juste nécessaires à l'accomplissement d'une tâche. Ce concept, défini initialement pour les architectures matérielles, peut être élargi aux systèmes. Si les objets manipulés sont des lignes de niveaux, si les opérateurs sont choisis pour favoriser l'accumulation en vue d'une décision, si encore le souci permanent est la réutilisabilité des opérateurs et la cohérence des processus de décision appliqués, alors mes travaux de recherche sont en adéquation avec ce concept de vision fruste. L'ensemble de mes activités de recherche peut être ainsi abordé selon deux points de vues :

- L'applicatif met en évidence les thèmes de complexité croissante progressivement traités : détection et estimation du mouvement en caméra fixe, recalage d'images en caméra mobile (type de mouvement connu et profondeur des objets contrainte) puis estimation générale du mouvement propre et de la structure de la scène en caméras embarquées sur un véhicule mobile.

- Le systémique met en exergue les objets et mécanismes privilégiés dans le concept de vision que je défends.

Je présenterai mes résultats majeurs en développant ci-dessous les approches proposées pour chaque thème selon le premier point de vue. Le deuxième permettra de mieux appréhender mes perspectives.

Détection et estimation du mouvement en caméra fixe : une difficulté majeure dans ce domaine est d'extraire des primitives image insensibles aux perturbations d'intensité entre images successives, en particulier les variations d'illumination. J'ai proposé la

construction de primitives géométriques locales, basées sur les lignes de niveaux pour s'affranchir des niveaux de gris eux même et ne garder que les relations d'ordre entre niveaux de gris. Cette méthode, utilisée en détection du mouvement, a été appliquée et évaluée sur plusieurs applications en vidéosurveillance : détection de véhicules et piétons mobiles (ma thèse à l'INRETS), détection et estimation de mouvement de foules dans le métro (projet Européen CROMATICA), détection de présence sur les ponts mobiles (mon emploi d'ingénieur en traitement d'images à CITILOG) et détection d'unicité de présence dans des salles bancaires critiques (contrat de prestation entre l'IEF et la société PMC).

Recalage d'images : l'hypothèse est ici celle d'un capteur dont le modèle de mouvement est connu ou approché. De plus, les objets sont à profondeur constante ou assez éloignés (i.e. des hypothèses sur la structure de scène). Pour fournir des éléments de réponse aux problèmes de variations d'éclairement et d'ambiguïtés d'appariement, nous retenons des primitives construites à partir des lignes de niveaux, robustes aux variations de contraste et adaptées par leur redondance à une stratégie d'appariement type « vote ». L'abondance de lignes de niveaux dans l'image favorise une mise en correspondance cumulative où chaque ligne est appelée au vote à une phase donnée en fonction de sa fiabilité. L'espace de vote est restreint par les listes de préférences d'une technique de décision sur graphe bi-partite dite des « mariages stables » [GI89]. Nous appliquons l'approche au recalage de profils 3D (ex. peigne de nano capacités dans un accéléromètre type MEMS) en microscopie électronique à balayage de franges (collaboration avec le département Minasys, IEF). L'application est particulièrement intéressante par la présence de nombreux motifs répétitifs dans ces images. Nous avons comparé notre approche avec 3 techniques classiques : corrélation de phases, Levenberg-Marquardt et approche différentielle de UC Santa Barbara. Aucune des trois ne donne des résultats probants. Notre appariement cumulatif priorisé de lignes de niveaux est concluant. Ceci incite à le généraliser :

1. à des transformations 3D complexes (thèse de Y. Almehio). On recherche des invariants (rapport de longueurs, coordonnées barycentriques, birapports, etc.) de primitives plus complexes. Selon la transformation décrivant le déplacement par stéréo ou mouvement, les primitives groupent un nombre de segments de lignes de niveaux croissant avec le nombre de paramètres du modèle. Pour une estimation de transformation type "translation 2D" sont construits des segments de lignes de niveaux, type "similarité" ce sont des angles (couples de segments), type « affinité » des Z et Y, type « projectif » des quadruplets de segments ;

2. à un appariement d'images ne préjugeant pas de l'application (reconnaissance, mouvement ou stéréovision) à base de jonctions robustes de lignes de niveaux (thèse de N. Suvonvorn). Le paradigme d'optimisation reste les mariages stables, mais ici optimisé pour une meilleure symétrie des populations et une satisfaction globale accrue. Exemple de résultat : une voiture sur la file opposée est détectée par notre véhicule PICAR en mouvement à 50 mètres au lieu de 20 obtenus jusque là avec un algorithme type Viterbi par lignes. Elle sera détectée 90% du temps contre 10% par la version précédente.

Estimation conjointe « ego-mouvement/structure de la scène » en caméras mobiles : **1)** une idée naturelle est de faire coopérer la stéréovision qui révèle la structure de la scène et le mouvement propre lui-même déterminé sur les séquences monoculaires. Dans la thèse d'A. Bak, l'extraction du mouvement propre est réalisée sur deux cartes de disparité et deux images successives. Des points d'intérêt robustes (SURF) sont extraits de chaque image. Le système d'équations instanciant la projection du mouvement est résolu selon un processus type RANSAC peu sensible aux faux appariements ou mouvements minoritaires. Cette méthode permet d'estimer les 6 paramètres du mouvement du capteur stéréo entre 2 poses pour ensuite extraire les objets dynamiques de la scène.
2) Si on contraint la nature du mouvement caméra (ex. translation) et les objets (approximation planaire) on identifiera les deux simultanément. En effet, parmi les nombreuses approches d'analyse 3D mobile de la littérature, deux attirent l'attention. Dans la première [FA95], il est montré que des vecteurs vitesse ayant des propriétés communes (module et/ou orientation) sont contraints d'appartenir à certaines courbes de l'image (sections coniques) dont les caractéristiques dépendent uniquement des paramètres 3D du mouvement. Dans la seconde [LAT02], les auteurs proposent une technique basée sur la stéréovision et le concept novateur de *v-disparité*, exploitant la proportionnalité entre la disparité et les indices ligne des images rectifiées. Ces deux études rejoignent nos propres conclusions : elles exploitent des courbes iso-valeur, de vitesse ou de disparité. De plus, la *v-disparité* est "cumulée" pour faire émerger une structure. Nous avons alors montré que la *v-disparité* n'est qu'un cas particulier d'un paradigme de vision général. Pour cela, nous l'avons généralisée au mouvement 3D avec la définition d'un espace de vote appelé *c-velocité* par analogie. Les "vitesses apparentes" 2D sont cumulées le long de courbes où elles sont constantes, au moins fortement majoritaires, si les pixels correspondants appartiennent à une surface de type particulier (ex. plan vertical ou horizontal). Dans l'espace de vote ainsi structuré, les objets paramétrés ressortent comme des courbes définies (ex. plan 3D = parabole). La technique a été appliquée avec succès à la détection d'obstacles à partir d'une caméra embarquée dans le cas où le déplacement du véhicule est majoritairement translationnel. Elle ne nécessite aucune calibration préalable et aucune connaissance *a priori* du mouvement propre de la caméra, mouvement en revanche vérifiable *a posteriori*. Les premières expérimentations sur cette approche confirment la robustesse recherchée : malgré une mauvaise qualité d'estimation des vitesses (un flot optique basique) et la présence de rotations d'amplitudes non négligeables, nous obtenons bien les résultats escomptés.

Mes perspectives de recherche sont motivées par le point de vue systémique que j'ai délibérément adopté. L'idée initiale est que l'autonomie d'un système implique, ne serait-ce que pour raisons énergétiques, une faible variété d'opérateurs de perception, dont les algorithmes de vision (segmentation et décision). Les "primitives" extraites des images seront intrinsèquement robustes et stables vis-à-vis de perturbations variées. Elles doivent de plus anticiper, voire faciliter, un processus de décision voulu systématique aux divers stades. Les lignes de niveaux répondent parfaitement à ces contraintes : on vérifie sans peine leur robustesse et leur abondance dans une image suggère et alimente un processus de décision cumulatif (i.e. manipulant un objet unique, l'histogramme). Nos travaux se déclinent alors en trois catégories de cumul, chacun associé de manière réconfortante à un stade de l'analyse d'images.

1 Au plus bas niveau, nous retenons l'information binaire apparition/disparition d'une primitive dans le temps. La complexité se situe strictement sur l'axe temporel. Le cumul dans le temps nous permet ainsi de reconstruire la scène fixe et donc par soustraction du fond, l'image des objets mobiles. Les espaces de vote sont 1D et multiples, affectés à chaque primitive.

2 Le consensus se voudrait spatio-temporel au deuxième niveau pour identifier le mouvement. Il restera d'abord spatial en pratique pour raisons de complexité : des primitives voisines dans l'image s'associent pour former des "pré-objets" contraints exhibant ainsi des invariants exploitables : leur mouvement à instancier doit être cohérent. Le cumul s'opère donc cette fois selon un modèle de mouvement de la caméra. Les primitives votent pour la transformation globale qui les aurait conduites dans leur nouvelle position. L'espace de vote est commun à toutes les primitives et multidimensionnel (une dimension par paramètre de mouvement).

3 Au niveau le plus élevé, la sémantique accrue implique des hypothèses à la fois sur les primitives et sur l'origine du mouvement. Les primitives sont supposées appartenir à un même objet 3D (ex. un plan) présentant, pour un modèle de déplacement du capteur donné, une propriété caractéristique commune des vecteurs vitesse qui permet de l'extraire. En particulier, leurs amplitudes sont constantes le long de courbes image prédéfinies par leurs équations analytiques. Les primitives ne votent plus selon leur structure mais selon leur vitesse. Dans le cas d'une scène 3D approximée par un ensemble de plans et d'une caméra à mouvement majoritairement longitudinal, l'espace de vote (*c-velocité*) est bidimensionnel : une dimension pour la vitesse, l'autre pour le paramètre des courbes iso-vitesse. Chaque vitesse vote pour sa courbe. Les surfaces 3D émergent dans cet espace de vote comme courbes 2D connues (droites ou paraboles).

Ce fil conducteur guide la présentation de mes travaux de recherche : le chapitre 4 sera dédié aux primitives image issues des lignes de niveaux que nous avons proposées. J'expliquerai comment la robustesse de ces lignes nous a amenés à construire des primitives adaptées à des applications de complexité croissante. Le chapitre 5 sera consacré aux processus de décision cumulative proposés. Les approches envisagées seront classées en fonction de la nature et de la dimensionnalité des espaces de cumuls.

Chapitre 4

Des primitives image robustes à partir des lignes de niveaux

4.1 Introduction

S'inspirant de la théorie d'un phénoménologue gestaltiste, Gaetano Kanizsa, des mathématiciens du CEREMADE et de l'Université des Iles Baléares ont proposé en 1999 une théorie atomique de l'image [CCM99] afin de répondre à une préoccupation majeure des traiteurs d'images : quelles sont les informations atomiques fiables d'où devrait partir tout algorithme d'analyse ? Cette théorie a engendré un algorithme de simplification d'image qui met en évidence sa structure occlusive : il permet de retrouver les jonctions en T ou en X et d'établir une carte topographique qui exhibe toutes les lignes de niveaux de cette image. Les propriétés pertinentes des lignes de niveaux pour l'analyse d'images ont été ainsi parfaitement étudiées. Elles se résument en trois points :

- Les lignes de niveaux sont insensibles vis-à-vis des variations de contraste globales dont les effets sont un changement des niveaux sans effet sur l'ordre relatif [1]. Les effets des changements de contraste sur les lignes de niveaux ont été étudiés en détails dans [MG00] et [Bou98].
- La représentation d'une image par lignes de niveaux est complètement inversible : on sait reconstituer l'image à partir de sa représentation.
- L'existence d'une définition mathématique unique autorise une étude rigoureuse des propriétés et une comparaison claire et sans ambiguïtés avec d'autres primitives éventuelles. A l'opposé, notons par exemple l'absence de définition mathématique de la notion de contour, dont la conséquence immédiate est la multiciplité des détecteurs - tous plus ou moins combinés avec des opérateurs de lissage - rendant la comparaison malaisée.

Notre choix s'est très tôt porté sur l'utilisation des lignes de niveaux dont nous avons montré expérimentalement la robustesse [Bou98]. Les applications sur lesquelles nous nous sommes focalisée cette dernière décennie ne pouvaient que tirer bénéfice de ce choix initial.

1. Toute fonction croissante au sens large peut être considérée comme une variation globale de contraste.

En effet, en vision dynamique comme en stéréovision ou en recalage d'images, le choix des primitives initiales est crucial : le mouvement ou le recalage ne peuvent être déterminés qu'en comparant les images d'une séquence. De même, l'étape d'appariement stéréoscopique se base sur la comparaison des images issues de deux caméras, fournissant chacune un point de vue différent d'une même scène. Cette comparaison doit, de ce fait, s'appuyer sur des descripteurs associés à des primitives invariantes dans le temps et l'espace (point de vue).

Pour répondre à cette préoccupation, une multitude de primitives avec leurs descripteurs ont été proposées dans la littérature. Parmi d'autres, citons : le détecteur multi-échellle d'Harris [MS04], invariant par rapport à la rotation, qui met en évidence les "coins", maxima locaux d'une fonction d'intérêt calculée à partir de la matrice d'auto-corrélation, estimée à différentes échelles ; les points SIFT [Low04] qui sont des *extrema* locaux spatiaux et multi-échelle de filtres DoG, ils correspondent à des pics et vallées lors du processus de lissage de l'image, chaque point extrait est alors associé à une échelle et une orientation ; le détecteur multi-échelle SURF [BETG08], plus rapide, qui utilise une collection de filtres de Haar afin d'approximer les dérivées secondes à plusieurs échelles, les points SURF sont définis comme étant les maxima locaux du déterminant de la matrice Hessienne ou enfin le détecteur MSER [MCUP02] qui utilise une segmentation afin de sélectionner les super pixels : répondant à une certaine propriété d'invariance et de stabilité, ils correspondent à des extrema régionaux de taille donnée stables vis-à-vis des transformations perspectives.

Notre choix pour les lignes de niveaux n'a pas été remis en cause malgré l'apparition et le regain d'intérêt pour ces "nouvelles" primitives. En effet, on constatera que les critères qui nous ont amenés à faire ce choix ne sont pas automatiquement vérifiés pour ces dernières. En particulier : la non dépendance directe aux niveaux de gris (mais la prise en compte plutôt des relations d'ordre entre les niveaux) ; la distinction explicite par le processus d'extraction entre l'information "contraste" et l'information "géométrie" ou "structure" dans l'image ; la non utilisation de seuils d'extraction difficiles à ajuster et dépendants de l'image. En effet, nous avons pris le parti de bannir les seuils lors de l'extraction des lignes de niveaux. De ce fait, toutes les lignes de niveaux sont détectées afin de ne perdre aucune information utile à la compréhension des structures géométriques. Cependant, ce choix a pour conséquence directe le très grand nombre de primitives à manipuler. Ce qui aurait pu constituer un inconvénient majeur à l'utilisation des lignes de niveaux est alors tourné en avantage puisque cela autorise un processus de décision cumulatif où la taille de la population des lignes supporte l'émergence d'une décision dès que nécessaire. En revanche, nous sommes conduits à définir la notion de fiabilité d'une ligne, chacune d'elle étant ainsi utilisée à une étape adéquate de la décision.

Dans ce chapitre, nous revenons sur le processus d'extraction des lignes de niveaux que nous avons proposé. En raison des applications considérées, nous avons fait le choix d'une extraction et d'une description locales des lignes. Par ailleurs, nous nous intéressons aussi bien à la détection d'objets en mouvement avec caméra fixe ou mobile qu'au cas de deux caméras stéréoscopiques. Dans chacun des cas, nous proposons alors des primitives issues des lignes de niveaux adaptées :

- Dans le cas où la caméra est fixe, l'objectif est la détection ponctuelle d'objets en mouvement. Nous proposons alors une extraction des lignes en chaque point. Les descripteurs retracent les orientations locales des lignes extraites.
- Dans le cas du recalage d'images (caméra en mouvement de translation planaire avec rotation autour de l'axe optique et zoom éventuel : transformation affine, homographie), le modèle de transformation étant connu et les invariants de cette transformation parfaitement définis. Nous proposons alors de définir des "segments" de lignes de niveaux (faisceau rectilignes de lignes de niveaux) et de les grouper en "V", "X", "Y" ou "Z" afin de pouvoir calculer les invariants adéquats.
- Lorsque la transformation n'est pas globale (caméra embarquée et/ou mouvement général), lorsque le déplacement d'une primitive dépend de sa profondeur (stéréovision), la primitive ne peut être que ponctuelle (la plus locale possible) mais avec toutefois une nécessité de définir des points les plus stables et robustes possibles à partir des lignes de niveaux. Nous proposons alors les "jonctions" de lignes de niveaux.

4.2 Un processus simple et efficace d'extraction des lignes de niveaux

Soit $I(\mathbf{p})$ l'intensité lumineuse du pixel de coordonnées $\mathbf{p}(x, y)$. L'ensemble de niveaux $\aleph_\lambda^I$ est l'ensemble de tous les points de l'image I ayant une luminosité supérieure ou égale à λ : $\aleph_\lambda^I = \{\mathbf{p} / I(\mathbf{p}) \geq \lambda\}$. Les ensembles de niveaux sont une représentation complète de l'image puisqu'il est possible de la reconstituer à partir des $\aleph_\lambda^I$. En effet, $I(\mathbf{p}) = \sup\{\lambda / I(\mathbf{p}) \geq \lambda\}$. La frontière d'un ensemble de niveaux est appelée "ligne de niveaux". La représentation par lignes de niveaux a été étudiée en détails par V. Caselles, J.M. Morel et J. Froment [CCM99], [Fro99]. Notons en particulier sa propriété d'invariance par rapport aux changements de contraste et sa commutation avec une transformation affine (translation, rotation, zoom). Caselles propose alors de considérer les lignes de niveaux comme les atomes de base sur lesquels devrait s'appuyer tout algorithme d'analyse d'images.

L'extraction des lignes de niveaux est souvent réalisée à partir des ensembles de niveaux associés en effectuant simplement une série de seuillages. L'approche que nous proposons permet une exploitation immédiate et efficace du résultat en fournissant directement des faisceaux rectilignes de lignes de niveaux superposées. En effet, nous exploitons une propriété élémentaire des ensembles de niveaux que traduit leur inclusion : $\forall \lambda > \mu \quad \aleph_\lambda^I \subset \aleph_\mu^I$. Par conséquent, les lignes de niveaux peuvent se superposer mais jamais se croiser. Nous proposons un simple processus récursif de suivi de lignes par groupe (se superposant), qui s'arrête lorsque le groupe cesse d'être rectiligne. On peut vérifier que la complexité totale n'est pas supérieure à celle d'une détection puis codage de lignes sur multi-seuils. De plus, notre processus n'utilise à aucun moment les niveaux de gris directement mais seulement leur ordre relatif, ce qui le rend en lui même particulièrement robuste aux variations de contraste. Il démarre en chaque point [2] $\mathbf{p}_o(x, y)$, détermine lequel parmi ses 4 voisins $\mathbf{p}_o(x, y+1)$, $\mathbf{p}_o(x-1, y)$, $\mathbf{p}_o(x, y-1)$ ou $\mathbf{p}_o(x+1, y)$ est le successeur, selon les 4

2. En réalité, les lignes de niveaux passent **entre** les pixels. Le référentiel est donc placé au niveau de l'espace inter pixel.

chemins possibles (voir FIGURE 4.1). Chaque successeur sélectionné devient à son tour le point courant p_k et le processus se répète jusqu'à ce que l'un des critères d'arrêt, définis dans les conditions ci-dessous, cesse d'être vérifié.

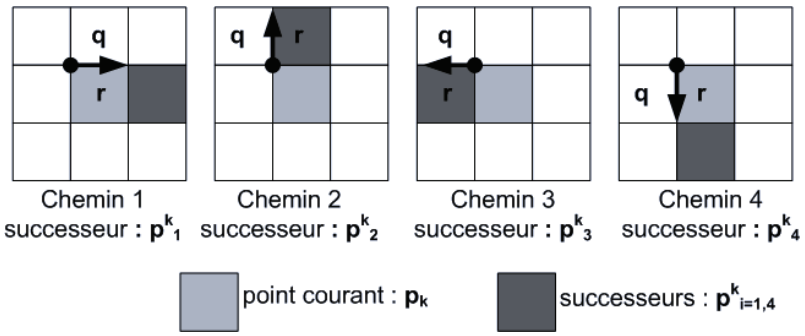


FIGURE 4.1 – Le point courant et ses 4 successeurs possibles en fonction des chemins 4-connexité.

$\mathbf{p}^k_{i=1,4}$ est un successeur de $\mathbf{p}_k$, à l'étape k si les cinq conditions suivantes sont vérifiées :

Condition 1 Au moins une ligne de niveau passe entre les pixels $\mathbf{q}$ et $\mathbf{r}$: $|I(\mathbf{r}_k) - I(\mathbf{q}_k)| \geq$ *seuil*.

Dans le but de prendre en compte les effets de quantifications, le seuil de différence de niveaux ci-dessus est fixé à 2 (pas de quantification + 1). Par conséquent, toutes les lignes sont extraites à ce stade sans aucune présélection. Il est à noter que le processus d'extraction ne nécessite à aucun moment de fixer des seuils arbitraires.

Condition 2 Les lignes de niveaux suivies se poursuivent effectivement :
$\left[\min\left(I(\mathbf{q}_{k-1}),\ I(\mathbf{r}_{k-1})\right),\ \max\left(I(\mathbf{q}_{k-1}),\ I(\mathbf{r}_{k-1})\right)\right] \cap$
$\left[\min\left(I(\mathbf{q}_k), I(\mathbf{r}_k)\right),\ \max\left(I(\mathbf{q}_k), I(\mathbf{r}_k)\right)\right] \neq \emptyset$

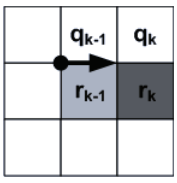


FIGURE 4.2 – Condition de test de la poursuite effective d'un groupe de lignes de niveaux.

Cette condition permet de garantir le suivi du même groupe de lignes (voir FIGURE 4.2). Certaines lignes peuvent toutefois se séparer du groupe en empruntant des chemins différents. Le processus de suivi étant récursif, chaque nouveau chemin engendré est aussi exploré tant que la condition 2 reste vérifiée. Cette dernière garantit en fait la non confusion des lignes et le respect de la propriété d'inclusion des ensembles de niveaux.

Condition 3 L'intérieur (resp. extérieur) des ensembles de niveaux – associés au groupe de lignes de niveaux suivi – est toujours laissé du même côté.

Ce qui se traduit par : $I(\mathbf{r}_k)$ est toujours plus grand (resp. plus petit) que $I(\mathbf{q}_k)$ pour le même groupe de lignes de niveaux. Cette condition évite ainsi les retours en arrière.

Condition 4 Les lignes de niveaux suivies sont toujours rectilignes.

Cette condition est testée en calculant la distance maximale entre les points déjà parcourus et la corde formée par le point initial et le point courant. Cette distance ne doit pas dépasser un seuil. Les points déjà parcourus ne sont bien sûr pas stockés. Seul le code de Freeman 4-connexité correspondant au chemin parcouru entre deux points successifs est gardé à chaque étape. L'origine du référentiel - pour le calcul de l'équation de droite - est le point initial $\mathbf{p}_o$.

Condition 5 La longueur maximale en pixels du trajet parcouru est atteinte.

Cette dernière condition peut être utilisée en remplacement de la condition 4 dans certaines applications (détection de mouvement). Le processus itératif s'arrête lorsque l'une des conditions ci-dessus n'est plus vérifiée. Il est possible alors d'associer au point initial $\mathbf{p}_o$, à l'issue de cette étape, les caractéristiques suivantes :

- le milieu $\mathbf{p} = \frac{\mathbf{p}_o + \mathbf{p}_k}{2}$ plus stable que les extrémités ;
- l'orientation approchée de ce groupe de lignes rectilignes $\theta_{\overrightarrow{\mathbf{p}_o\mathbf{p}_k}}$;
- le contraste moyen de part et d'autre des lignes suivies : $c = \frac{\sum |I(\mathbf{r}_k) - I(\mathbf{q}_k)|}{k}$;
- et enfin la longueur du trajet : $l = \left\| \overrightarrow{\mathbf{p}_o\mathbf{p}_k} \right\|$.

Le processus d'extraction des lignes de niveau fournit directement en sortie un ensemble de **segments** de lignes de niveau, qui constituent les entités de base du groupement décrit dans la section 4.3, $\left\{S_I^i\right\}_{i=1,\eta}$ associés à l'image I avec leurs caractéristiques, :

$$\overrightarrow{S_I^i} = \left[\begin{array}{cccc} \mathbf{p}_i & \theta_i & l_i & c_i \end{array} \right]^T.$$

Définition 1 *Un "segment" désigne un groupe de lignes de niveaux rectilignes obtenu à partir de la procédure de suivi.*

Définition 2 $F_{p,u,v} = \{L_\lambda / \lambda \in]u, v]\}$ *est un flux défini localement au point p dans l'image. v et u sont les lignes de niveaux supérieures et inférieures du flux respectivement.*

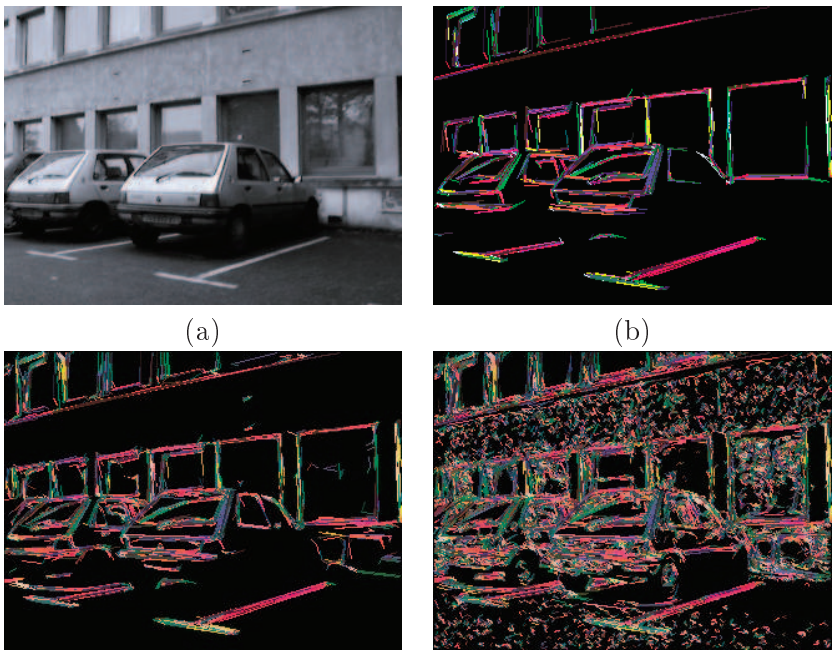
Définition 3 *Le nombre de lignes de niveaux associé à un flux*[3] *est appelé "quantité de flux" :* $\mathcal{E} = v - u$.

Fiabilité des lignes de niveaux

Les lignes de niveaux n'ont pas toutes la même importance visuelle et ne coïncident pas toujours avec les contours que nous percevons à l'oeil nu. En effet, certaines séparent des ensembles de niveaux en réalité très proches en terme de niveaux de gris. Par ailleurs, les *segments* de faible longueur sont probablement associés à des ensembles de niveaux de

3. Notons que des flux peuvent se séparer ou se rejoindre à n'importe quel endroit de l'image.

taille réduite générés par du bruit. Il existe plusieurs moyens de sélectionner les lignes de niveaux qui correspondent le mieux aux contours perçus. Citons à ce propos les travaux de Jacques Froment [Fro99] ou ceux de N. Paragios et R. Deriche [PD98]. Nous avons choisi, dans le cadre des applications que nous traitons, de tirer profit des différences perceptuelles entre les lignes extraites afin de définir une notion de fiabilité basée sur la longueur l du *segment* et le contraste moyen c de part et d'autre de chaque segment. Les *segments* jugés peu fiables ne seront pas pour autant complètement écartés du processus de décision : ils y contribuent avec toutefois un plus faible crédit. Afin de préparer cette étape, nous avons choisi de classer les *segments* $\left\{L_I^i\right\}_{i=1,\eta}$ par ordre décroissant (des plus fiables au moins fiables) du produit $l \times c$, considéré comme étant un indice de fiabilité (voir FIGURE. 4.3). Par ailleurs, nous proposons une analyse plus fine de la répartition statistique de l'indice de fiabilité permettant de découper les primitives en plusieurs catégories. Ce procédé sera détaillé dans le chapitre suivant consacré aux processus de décision cumulatifs.



(a) (b)

(d) (d)

FIGURE 4.3 – Exemple de classement des segments des plus fiables aux moins fiables. a) Image originale extraite d'une séquence prise par une caméra embarquée (Parking IEF). b) Les $\frac{\eta}{4}$ premiers segments qui correspondent aux plus fiables. Le nombre total de segments $= \eta$. Les segments sont retracés sur l'image en utilisant les caractéristiques trouvées (longueurs, orientation, point milieu). c) Les $\frac{\eta}{4}$ suivants (moins fiables). d) Les $\frac{\eta}{4}$ suivants (encore moins fiables). etc.

4.3 Configurations de segments guidées par un modèle de déformation de l'image

Nous commençons notre étude en nous intéressant à la hiérarchie des transformations géométriques en termes de complexité. Ainsi, on définit des primitives pouvant être construites de manière progressive de la plus simple à la plus complexe. Le système qui les exploite est alors capable de fournir une transformation réponse quel que soit le temps de calcul alloué. Nous devons tenir compte du nombre de paramètres et des invariants de la transformation considérée (voir TABLE 4.1 et 4.2). Le nombre de paramètres nous permet de déterminer le nombre de points caractéristiques au sein d'une même primitive nécessaires à l'estimation de la transformation, il correspond de fait au nombre d'équations à résoudre. Nous focaliserons nos explications dans cette section sur le procédé de groupement proprement dit, développé dans la thèse de **Yasser Almehio**, visant à former des "primitives".

Définition 4 *Une "primitive" désigne un ensemble de p segments configurés de manière pertinente en fonction du modèle de déformation de l'image.*

Transformations	Euclidienne	Similarité	Affine	Projective
Rotation	X	X	X	X
Translation	X	X	X	X
Changement d'échelle uniforme		X	X	X
Changement d'échelle non uniforme			X	X
Cisaillement			X	X
Projection perspective				X

TABLE 4.1 – Hiérarchie des transformations géométriques

Transformations	Euclidienne	Similarité	Affine	Projective
Longueur	X			
Angle	X	X		
Rapport de longueurs	X	X		
Parallélisme	X	X	X	
Coïncidence	X	X	X	X
Bi-rapport	X	X	X	X

TABLE 4.2 – Invariants associés aux transformations géométriques

Cas des transformations Euclidienne et de Similarité : Ces transformations admettent respectivement 3 et 4 paramètres. Dans les deux cas, deux points caractéristiques sont nécessaires, chaque point fournissant deux coordonnées. Ces deux points peuvent être fournis simplement par deux segments que nous proposons de grouper en primitive "angle".

Définition 5 *Un "angle" désigne un ensemble de $p = 2$ segments.*

Soit deux segments $\overrightarrow{S_I^i}$ et $\overrightarrow{S_I^j}$ et leurs vecteurs de caractéristiques respectifs :

$$\overrightarrow{S_I^i} = \left[\begin{array}{cccc} \mathbf{p}_i & \theta_i & l_i & c_i \end{array} \right]^T \text{ et } \overrightarrow{S_I^j} = \left[\begin{array}{cccc} \mathbf{p}_j & \theta_j & l_j & c_j \end{array} \right]^T$$

L'angle formé par ces deux segments est alors :

$$\overrightarrow{A_I^{i,j}} = \left[\begin{array}{cccccc} \mathbf{p}_i & \mathbf{p}_j & \triangle\theta & \ell & c_i & c_j \end{array} \right]^T$$

avec $\triangle\theta = \theta_i - \theta_j$ et $\ell = \frac{l_i}{l_j}$ les invariants des transformations Euclidienne et de Similarité. Ce groupement est valide ssi deux conditions sont vérifiées (voir FIGURE 4.4) :

1. Les segments considérés ne sont pas colinéaires : $\triangle\theta > \triangle\theta_{\min}$ ($\triangle\theta_{\min}$ doit être fixé en fonction de la courbure maximale autorisée dans le processus d'extraction).
2. Les segments considérés sont suffisamment proches en terme de distance : $\left\|\overrightarrow{\mathbf{p}_i\mathbf{p}_j}\right\| < \mathbf{dist}_{\max}$. Le choix de $\triangle\theta_{\min}$ et $\mathbf{dist}_{\max}$ est conditionné par l'amplitude maximale attendue (de la transformation) afin d'éviter la disparition de segments entre les deux images.

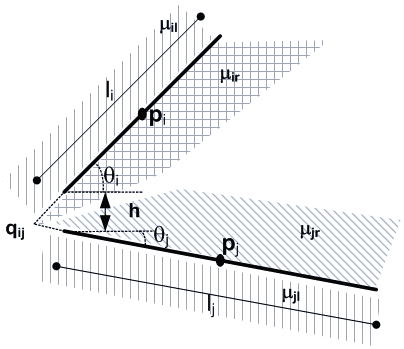


FIGURE 4.4 – Le couplage des segments pour former une primitive "angle" s'appuie sur les conditions de non colinéarité et de proximité.

Cas de la transformation affine : La transformation affine est caractérisée par 6 paramètres. Les segments doivent donc être groupés en triplets dont les formes sont construites par une extension de la primitive "angle".

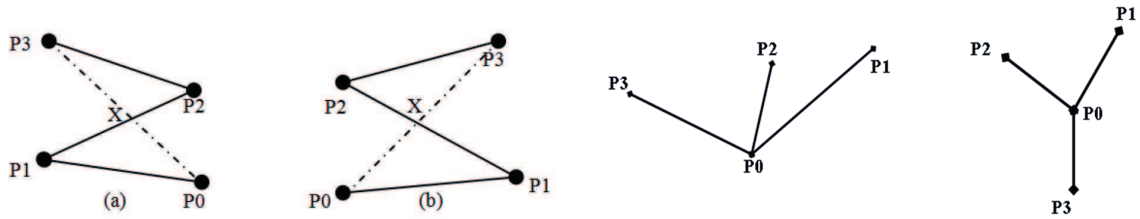
Définition 6 *Un "Z" ou un "Y" désigne un ensemble de $p = 3$ segments groupés sous la forme de Z ou Y. Ils sont construits à partir d'un angle auquel on ajoute un segment.*

Soit quatre points P_0, P_1, P_2 et P_3 formant 3 segments $\overrightarrow{S_I^i}$, $\overrightarrow{S_I^j}$, $\overrightarrow{S_I^k}$ et $\overrightarrow{S_I^l}$. Soit X le point d'intersection entre les segments $[P_0, P_3]$ et $[P_1, P_2]$ tel que décrit dans la FIGURE 4.5 (a). A partir de ce "Z", deux invariants ρ et σ sont définis :

$$\rho = \frac{\|P_3X\|}{\|P_0X\|} \text{ et } \sigma = \frac{\|P_2X\|}{\|P_1X\|} \text{ avec } \rho \geq 1.$$

Dans le cas d'un groupement en "Y", voir FIGURE 4.5 (b), les invariants α, β et γ vérifient :

$$\begin{cases} \alpha P_1 + \beta P_2 + \gamma P_3 &= P_0 \\ \alpha + \beta + \gamma &= 0 \end{cases}$$



(a) (b)

FIGURE 4.5 – Groupement en Z et en Y à partir de trois segments

Cas de la transformation projective : La transformation projective est caractérisée par 8 paramètres. 4 points colinéaires sont nécessaires. Ces points sont issus du groupement de 4 segments ou plus précisément d'un Z ajouté à un segment. Le "Z" étendu (en ajoutant un segment supplémentaire) donne un "W" bien défini par 4 segments (voir FIGURE 4.6).

Définition 7 *Un "W" désigne un ensemble de $p = 4$ segments groupés sous la forme de W. Ils sont construits à partir d'un "Z" ajouté à un segment.*

Soit quatre points P_0, P_1, P_2 et P_3 issus de 4 segments. Le bi-rapport est défini de la manière suivante :

$$B(P_0, P_1, P_2, P_3) = \frac{\|P_0P_1\|}{\|P_0P_2\|} \frac{\|P_2P_3\|}{\|P_1P_3\|}$$

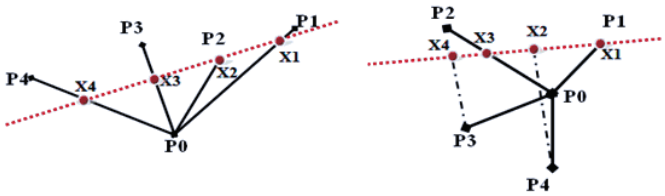


FIGURE 4.6 – Groupement en W à partir de quatre segments

Récapitulatif et exemple de résultats : La TABLE 4.3 résume le type de primitives obtenu par groupement en fonction de la nature de la transformation recherchée. Elle donne de plus le nombre de paramètres de la transformation, justifiant ainsi le nombre de segments à grouper nécessaires pour former une primitive.

Dans les images de la FIGURE 4.7, des exemples de primitives de type "Z", "Y" et "W" extraites sont donnés. Il est à noter que le choix d'enrichir des primitives de base

Transformations	Euclidienne	Similarité	Affine	Projective
Nombre de paramètres	3	4	6	8
Nombre de points caractéristiques	2	2	4	4
Primitives construites	V	V	Z ou Y	W

TABLE 4.3 – Groupement des lignes de niveaux en fonction des transformations

pour "fabriquer" des primitives plus complexes nous a naturellement conduit à grouper des segments connectés. Il aurait été envisageable de relâcher cette contrainte mais au prix d'une complexité plus importante liée à l'augmentation de la zone de recherche des segments candidats au groupement.

(a) (b) (c)

FIGURE 4.7 – Exemple de Z, Y et W extraits

4.4 Extraction directe des jonctions de lignes de niveaux

Lorsque les déformations entre images sont locales ou lorsque le modèle de ces déformations n'est pas présupposé, quelle que soit la raison, nous proposons de construire des primitives générales basées sur la topologie locale dans l'image, que nous appelons "jonctions de lignes de niveaux". L'approche d'extraction de ces jonctions est basée sur les groupes de lignes de niveaux superposés précédemment définis, appelés "flux de lignes de niveaux" $F_{p,u,v}$, par souci de concision. Pour renforçer sa robustesse par rapport au bruit, un détecteur de variations d'intensités associé à un modèle appelé "EFLAM" (Extended Flow Laminating Average Milieu) est proposé. Inspiré du détecteur de Girard [Gir80], utilisé ensuite dans la méthode SUSAN [SB97], il est adapté pour détecter des variations d'intensité autours des jonctions et permet de ce fait de sélectionner les jonctions potentiellement fiables et stables. L'approche que nous proposons pour l'extraction des jonctions de lignes de niveaux exploite alors la variation d'intensité sur un voisinage puis l'extraction des flux. Décrivons plus précisément le principe de l'approche en commençant par définir la notion de "flux".

Rappelons à présent le principe de la détection de points d'intérêt par la méthode SUSAN. La variation d'intensité autours de chaque pixel k est déterminée en mesurant la similarité entre l'intensité du pixel considéré et celle de ses voisins, localisés dans un masque circulaire. $\mathcal{N}_k$ est le nombre des pixels, dont la dissimilarité - exprimée par la différence des

niveaux- est inférieure à un seuil $\mathcal{S}$.

$$\begin{aligned} \mathcal{N}_k &= \textstyle\sum_{k_i \in w} \mathcal{C}_{ki} \\ \mathcal{C}_{ki} &= \begin{cases} 1 & \text{si} \quad |I(k) - I(k_i)| \leq \mathcal{S} \\ 0 & \text{sinon} \end{cases} \end{aligned} \tag{4.1}$$

La région couverte par ces pixels est appelée USAN (Univalue Segment Assimilating Nucleus). Ainsi, si $\mathcal{N}_k$ est inférieur à $\frac{\mathcal{N}_m}{2}$ (où $\mathcal{N}_m$ est le nombre des pixels dans le masque), alors le pixel k est potentiellement considéré comme un point d'intérêt dont la variation de l'intensité $\mathcal{V}_k$ est définie par :

$$\mathcal{V}_k = \begin{cases} \frac{\mathcal{N}_m}{2} - \mathcal{N}_k & \text{si} \quad \mathcal{N}_k < \frac{\mathcal{N}_m}{2} \\ 0 & \text{sinon} \end{cases} \tag{4.2}$$

C'est sur un principe similaire que le modèle EFLAM a été établi. Supposons que la jonction se situe à un point $\tilde{p}$ entre quatre pixels quelconques $\tilde{p}_i$ dans l'image. $\tilde{F}_{\tilde{p},u,v}$ est le flux maximal passant à travers cette jonction avec la quantité $\mathcal{E}$. Si nous considérons la jonction sur des voisinages plus larges, la jonction au point $\tilde{p}$ est entourée par des régions homogènes. Comme pour SUSAN, ces régions sont donc déterminées, à partir des quatre pixels $\tilde{p}$, en mesurant leurs similarités :

$$\mathcal{N}_{\tilde{p}_i} = \textstyle\sum_{p_j \in w} \mathcal{C}_{p_j}, \quad o\grave{u}\ \mathcal{C}_{p_j} = \begin{cases} 1 & \text{si} \quad |I(\tilde{p}_i) - I(p_j)| \leq \frac{\mathcal{E}}{2} \\ 0 & \text{sinon} \end{cases} \tag{4.3}$$

Notons que le seuil utilisé est $\frac{\mathcal{E}}{2}$ où $\mathcal{E}$ est le nombre maximal des lignes de niveaux au point $\tilde{p}$. La région R_i couverte par des pixels C_i est appelée le FLAM.

$$R_{\tilde{p}_i} = [\ p_j \in \tilde{p}_i / \mathcal{C}_{p_j} = 1\ ] \tag{4.4}$$

Puisque la jonction est considérée ainsi à partir d'un voisinage, il est alors nécessaire de recalculer le flux passant à travers la jonction en fonction de ses voisins. Pour cela, nous déterminons d'abord l'intensité moyenne pour chaque régions R_i :

$$\mathcal{M}_{\tilde{p}_i} = \frac{\sum_{[\ p_j \in \tilde{p}_i / \mathcal{C}_{p_j} = 1\ ]} I(p_j)}{\mathcal{N}_{\tilde{p}_i}} \tag{4.5}$$

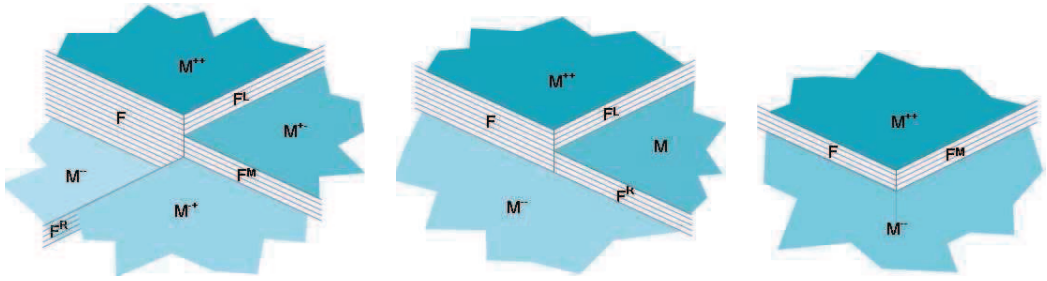
La FIGURE 4.8 montre le schéma simplifié des trois types de jonction existantes dans l'image : la jonction de type X, de type T (ou de type Y), et de type L. La FIGURE 4.8 (a) présente une jonction de type X composée de trois flux dont $F^L_{\tilde{p},\mathcal{M}^{+-}_{\tilde{p}},\mathcal{M}^{++}_{\tilde{p}}}$, $F^M_{\tilde{p},\mathcal{M}^{+}_{\tilde{p}},\mathcal{M}^{+-}_{\tilde{p}}}$, et $F^R_{\tilde{p},\mathcal{M}^{--}_{\tilde{p}},\mathcal{M}^{-+}_{\tilde{p}}}$. Le flux maximal en moyenne passant au point $\tilde{p}$ est donc $F^*_{\tilde{p}} = F^L_{\tilde{p}} \cup F^M_{\tilde{p}} \cup F^R_{\tilde{p}}$. Notons que $\mathcal{M}^{++}_{\tilde{p}} \geq \mathcal{M}^{+-}_{\tilde{p}} \geq \mathcal{M}_{\tilde{p}} \geq \mathcal{M}^{-+}_{\tilde{p}} \geq \mathcal{M}^{--}_{\tilde{p}}$ où les intensités moyennes maximale et minimale sont $\mathcal{M}^{++}_{\tilde{p}}$ et $\mathcal{M}^{--}_{\tilde{p}}$ respectivement.

Ainsi, pour établir le modèle, nous commençons par retrouver le flux maximal en déterminant l'intensité moyenne $\mathcal{M}^{++}_{\tilde{p}}$ et $\mathcal{M}^{--}_{\tilde{p}}$. La direction du flux $F^*_{\tilde{p}}$ est alors définie

comme un flux entrant vers le point de jonction. Par conséquent, la relation d'ordre entre les intensités autours de la jonction associée à ce flux maximal pourra être déterminée. Nous déduisions alors l'intensité moyenne de $\mathcal{M}_{\tilde{p}}^{+-}$ et de $\mathcal{M}_{\tilde{p}}^{-+}$. Ainsi, les flux $F_{\tilde{p}}^{L}$, $F_{\tilde{p}}^{M}$, et $F_{\tilde{p}}^{L}$ pourrons être définis comme les flux sortants du point de jonction.

De la même façon, nous pouvons modéliser la jonction de type T (ou de type Y), composée par deux flux, voir FIGURE 4.8 (b).

Concernant le coin L, nous considérons cela comme une jonction lorsqu'il porte une variation importante comparée à celle des contours et des régions. La variation servira donc à distinguer des jonctions de type L et renforcera le potentiel des jonctions de type T et X.



(a) (b) (c)

FIGURE 4.8 – Schéma simplifié des jonctions suivant leur type. (a) jonction X. (b) jonction T(ou Y). (c) jonction L (coin).

La variation d'intensité autour des jonctions sera définie en utilisant les variations des régions concernées. La variation de chaque région $\mathcal{V}_{\tilde{p}_i}$ est définie de la manière suivante :

$$\mathcal{V}_{\mathcal{R}_i} = \begin{cases} \frac{\mathcal{N}_m}{2} - \mathcal{N}_{\tilde{p}_i} & \text{si} \quad \frac{\mathcal{N}_m}{16} \leq \mathcal{N}_{\tilde{p}_i} < \frac{\mathcal{N}_m}{2} \\ 0 & \text{sinon} \end{cases} \tag{4.6}$$

Cette variation est calculée comme dans la méthode SUSAN avec une modification sur le critère du seuil $\mathcal{N}_{\tilde{p}_i}$. Ce seuil doit être supérieur ou égal à $\frac{\mathcal{N}_m}{16}$. Il exprime la plus petite taille de région qui est acceptable comme un coin (la jonction de type L). Si $\mathcal{N}_{\tilde{p}_i}$ est plus petite que ce seuil, on le considère alors comme un bruit.

Nous pouvons donc définir la variation de la jonction $\mathcal{V}_{\tilde{p}}$ suivante :

$$\mathcal{V}_{\tilde{p}} = \begin{cases} \sum_{i\in[1,4]} \mathcal{V}_{\mathcal{R}i}^2 & \text{si} \quad \mathcal{M}_{\tilde{p}}^{++} - \mathcal{M}_{\tilde{p}}^{--} \geq \mathcal{E} \\ 0 & \text{sinon} \end{cases} \tag{4.7}$$

La variation est alors égale à la somme L_2 des variations de chaque région si l'écart entre les intensités maximale et minimale est supérieur ou égal à $\mathcal{E}$, sinon la variation est nulle. Comme précédemment, la valeur de $\mathcal{E}$ indique le nombre maximal de lignes de niveaux passant à travers la jonction (en moyenne sur ses voisinages). Partant de la contrainte sur $\mathcal{E}$ dans les équations (4.7) et (4.3), la variation acceptable à l'intérieur des régions doit être supérieure ou égale à $\mathcal{E}/2$.

Méthode d'extraction des jonctions

Première étape : Calcul des variations des jonctions selon le modèle EFLAM. La méthode (équation 4.7) est appliquée sur tous les points, entre quatre pixels quelconques, dans l'image. Les jonctions potentielles seront formées par les variations positives et non nulles. Le degré de variation donne une indication sur la fiabilité de la jonction. Notons que dans la méthode classique d'extraction des points d'intérêt, le point potentiel est déterminé localement par la suppression des variations non maximales. Dans notre méthode, ces suppressions seront basées sur la fiabilité des jonctions, qui ne dépend pas seulement de la variation, mais aussi des caractéristiques des flux et cela joue un rôle très important dans la robustesse. En effet, nous donnons ainsi une chance aux jonctions ayant une petite variation. Le flux maximal doit être supérieur ou égal à $\mathcal{E}$.

Deuxième étape : A chaque jonction potentielle, les intensités moyennes des régions sont déterminées, ainsi que les flux formés par la jonction. Par conséquent, le type de jonction sera déduit automatiquement. Notons que la jonction de type X peut être réduite à une jonction de type Y, et la jonction de type Y à une jonction de type L, par la suppression du plus petit flux, en fusionnant les régions des deux côtés. En général, la jonction de type X est rare et très sensible au bruit, ce qui nous amène alors à nous intéresser uniquement aux jonctions de type L et Y. Celles qui correspondent au type X doivent être déduites automatiquement du type Y. L'automate conçu pour traquer ces flux est alors celui décrit dans la section 4.2.

La primitive "jonction de lignes de niveaux" et ses descripteurs

Par définition, les jonctions de type L et Y sont composées par un et deux flux respectivement. Après avoir dépisté ces flux par l'algorithme présenté dans la section précédente, la primitive "jonction de lignes de niveaux" peut donc être caractérisée par les descripteurs suivants : les coordonnées du point de jonction, les segments, et les angles. Le vecteur descripteurs de la primitive est alors défini comme suit :

$$\overrightarrow{P_Y} = \left[\begin{array}{ccccc} \mathrm{p} & \overrightarrow{S}^* & \overrightarrow{S}^L & \overrightarrow{S}^R & \theta^L \quad \theta^R \end{array} \right] \ , \ \overrightarrow{P_L} = \left[\begin{array}{ccccc} \mathrm{p} & \overrightarrow{S}^* & \overrightarrow{S}^M & \theta^L & \theta^R \end{array} \right] \tag{4.8}$$

La jonction de type Y est caractérisée par une primitive à trois segments $\overrightarrow{S}^*$, $\overrightarrow{S}^L$, et $\overrightarrow{S}^R$ approximés à partir des flux $F^*_{\tilde{p},\mathcal{M}^{--}_{\tilde{p}},\mathcal{M}^{++}_{\tilde{p}}}$, $F^L_{\tilde{p},\mathcal{M}_{\tilde{p}},\mathcal{M}^{++}_{\tilde{p}}}$, et $F^R_{\tilde{p},\mathcal{M}^{--}_{\tilde{p}},\mathcal{M}_{\tilde{p}}}$ respectivement. Le point de jonction est le point de séparation des flux. Par rapport à ce point, la direction du segment $\overrightarrow{S}^*$, défini comme le segment principal de la primitive, sera la direction entrante. Les segments $\overrightarrow{S}^L$ et $\overrightarrow{S}^R$ auront des directions sortantes. Les angles θ^L et θ^R seront mesurés par rapport au segment principal. θ^L est l'angle entre le segment $\overrightarrow{S}^*$ et $\overrightarrow{S}^L$. θ^R est celui entre $\overrightarrow{S}^*$ et $\overrightarrow{S}^R$. La figure 4.9 montre la primitive associée à une jonction de type Y.

Notons qu'en définissant un segment principal pour la primitive, l'ordre de l'intensité des régions autour de la jonction ne changera pas. La primitive est ainsi invariante localement vis-à-vis de la rotation. Cette propriété s'avère très pratique pour tester la compatibilité entre les primitives dans le processus de mise en correspondance. Par ailleurs,

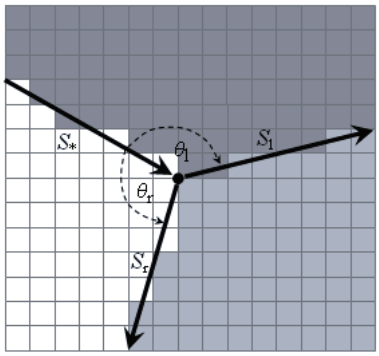


FIGURE 4.9 – Définition d'une primitive à partir d'une jonction.

les angles ne peuvent pas non plus changer de plus de 180 degrés. Concernant la jonction de type L, la primitive est définie de la même manière que la jonction de type Y. Notons cependant que la notion de segment principal de la primitive n'a plus grand sens.

$\mathcal{M}_k^l$ et $\mathcal{M}_k^r$ étant les intensités moyennes du côté gauche et droit du segment respectivement, si $[(\mathcal{M}_k^l, \mathcal{M}_k^r)/k \in [1 \cdots L]]$ est l'ensemble des intensités à chaque pas du segment, le contraste est alors formulé par :

$$C = \left\{ \frac{[\otimes\xi(\mathcal{M}^l)] \ominus [\otimes\xi(\mathcal{M}^r)]}{L} \right\} \oslash \left\{ \frac{\ominus[\xi(\mathcal{M}^l) \otimes \xi(\mathcal{M}^r)]}{L} \right\} \tag{4.9}$$

Où $\otimes$ $\ominus$ et $\oslash$ sont des opérateurs et $\xi(n_k)$ est une fonction pondérée à chaque pixel. Un exemple simple que nous employons est le suivant : $\otimes$, $\ominus$, et $\oslash$ sont les opérateurs d'addition, de soustraction, et de calcul du maximum respectivement. La fonction pondérée est définie par $\xi(\mathcal{M}_k) = \mathcal{M}_k$. Nous obtenons donc la fonction de contraste suivante :

$$C = \max \left\{ \frac{\sum_k \mathcal{M}^l - \sum_k \mathcal{M}^r}{L}, \frac{\sum_i (\mathcal{M}^l - \mathcal{M}^r)}{L} \right\} \tag{4.10}$$

Le contraste est finalement la valeur maximale de la différence des sommes ou de la somme des différences d'intensité.

La fiabilité d'une primitive est le produit de la fiabilité de ses segments $\mathcal{F}_S$ et de la variation des régions autour de la jonction $\mathcal{V}$: $\mathcal{F}_{\overrightarrow{P}} = \mathcal{V}_{\tilde{p}} \times \mathcal{F}_{\overrightarrow{S}}$

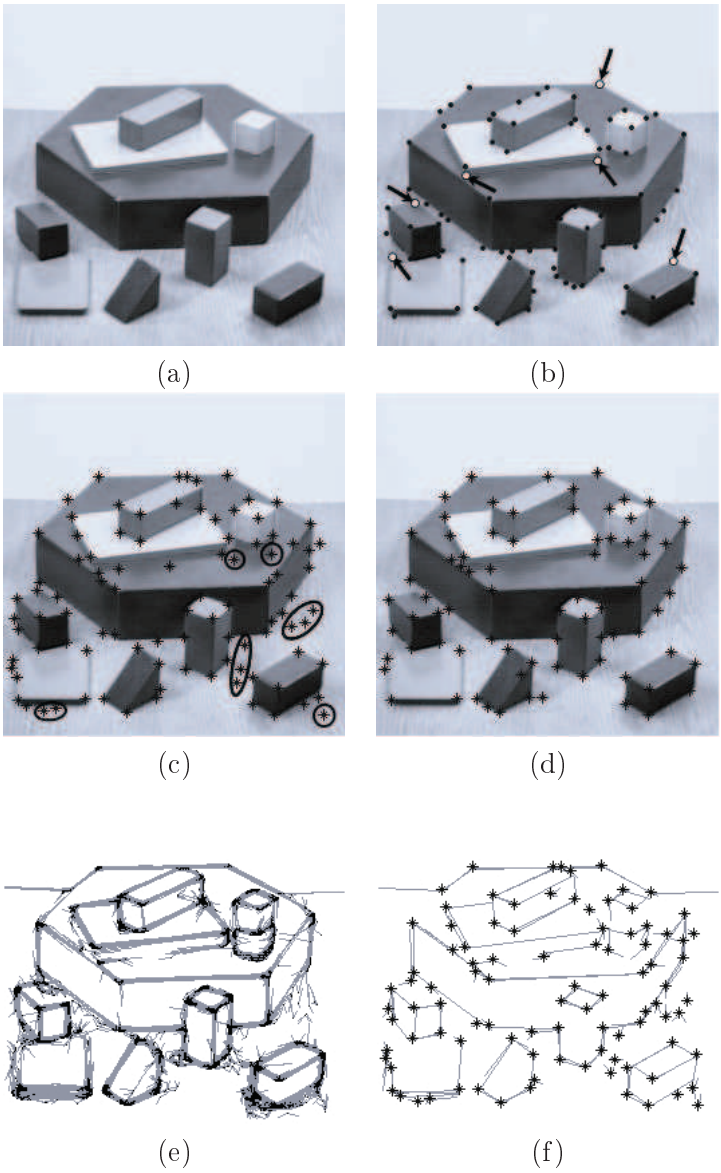
Expérimentation et discussion

Nous avons testé notre méthode en la comparant à celle de Harris sur deux images réelles, celle de la figure 4.10 (a) désignée par "bloc" et celle de la figure 4.11 (a) désignée par "house". Nous choisissons de la comparer au détecteur de Harris plutôt qu'à SUSAN ou tout autre approche car celui-ci est particulièrement adapté aux transformations affines même s'il n'est pas très précis ou robuste vis-à-vis des perturbations. Les meilleurs résultats de l'opérateur de Harris sont présentés en ligne dans "http ://www.cim.mcgill.ca/ dparks/" et montrés dans la FIGURE 4.10 (b) et 4.11 (b) respectivement. En appliquant notre méthode avec $\mathcal{E} = 10$, les jonctions extraites à partir de l'image "bloc" avant suppression

des non *maxima* sont montrées dans la FIGURE 4.10 (c). La FIGURE 4.10 (e) affiche les jonctions choisies parmi toutes les jonctions de la FIGURE 4.10 (c) en utilisant le critère de fiabilité. Les points de jonction sont alors superposés dans l'image "bloc" (FIGURE 4.10(c)) à comparer avec ceux obtenus par l'opérateur de Harris. Notons que les coins non détectés par la méthode de Harris sont distingués par des flèches. Nous constatons que notre méthode de détection est plus sensible aux ombres (voir les points cerclés dans la FIGURE 4.10 (c)). Nous constatons en revanche qu'en augmentant $\mathcal{E}$ à 14 (la FIGURE 4.10(f)), cela supprime les points associés à des ombres ou des contours. Pour ce qui concerne l'image "house", toutes les jonctions extraites puis sélectionnées sont montrées dans les FIGURES 4.11 (d) et 4.11 (e) respectivement. L'interprétation du résultat reste la même pour les FIGURES 4.11 (b) et 4.11 (c) ainsi que pour les FIGURE 4.10 (b) et 4.10 (c).

Ces résultats soulignent un avantage important de notre méthode : nous pouvons contrôler le nombre de jonctions extraites de l'image. Cela est très important par exemple pour la reconstruction 3D ou l'analyse du mouvement. L'extraction des jonctions sera exécutée comme un processus itératif en diminuant la valeur de $\mathcal{E}$. Au début, des jonctions sont extraites avec une certaine valeur de $\mathcal{E}$. Une zone d'exclusion est définie comme un secteur circulaire autour du point de la jonction extraite. A l'itération suivante, la méthode d'extraction sera encore une fois exécutée en utilisant un $\mathcal{E}$ plus petit, et ajoutant ainsi de nouvelles jonctions extraites seulement dans les zones autorisées. Notons que lorsque $\mathcal{E}$ est plus petit, le nombre de jonctions augmente. Puisque la fiabilité des primitives est calculée en utilisant le flux maximal à travers la jonction $\mathcal{E}$ en respectant la longueur et la variation de contraste, alors les primitives obtenues plus tôt dans le processus sont plus fiables. Cette propriété est fondamentale pour la méthode de mise en correspondance multi-échelle développée dans le cadre de la thèse de **Nikom Suvonvron**.

Les FIGURE 4.12 (a) et (b) illustrent un exemple d'images stéréoscopiques d'un buste. Les jonctions extraites des images gauche et droite à la première boucle, où $\mathcal{E} = 20$, sont montrées dans les FIGURES 4.12 (c) et (d) respectivement. Les jonctions extraites représentent les principales structures de l'image : la tête, les yeux, la bouche, et le nez. Les FIGURE 4.12 (c) et (d) retracent la deuxième boucle de l'itération, où $\mathcal{E} = 15$. De nouvelles jonctions s'ajoutent représentant mieux les détails du visage.



(a) (b) (c) (d) (e) (f)

FIGURE 4.10 – (a) L'image "bloc". (b) Les jonctions obtenues par le détecteur de Harris. (c) Les jonctions obtenues par notre méthode avec $\mathcal{E} = 10$. (d) Les jonctions obtenues par notre méthode avec $\mathcal{E} = 14$. (e) Les jonctions avant le filtrage avec $\mathcal{E} = 10$. (f) Les jonction après le filtrage avec $\mathcal{E} = 10$.

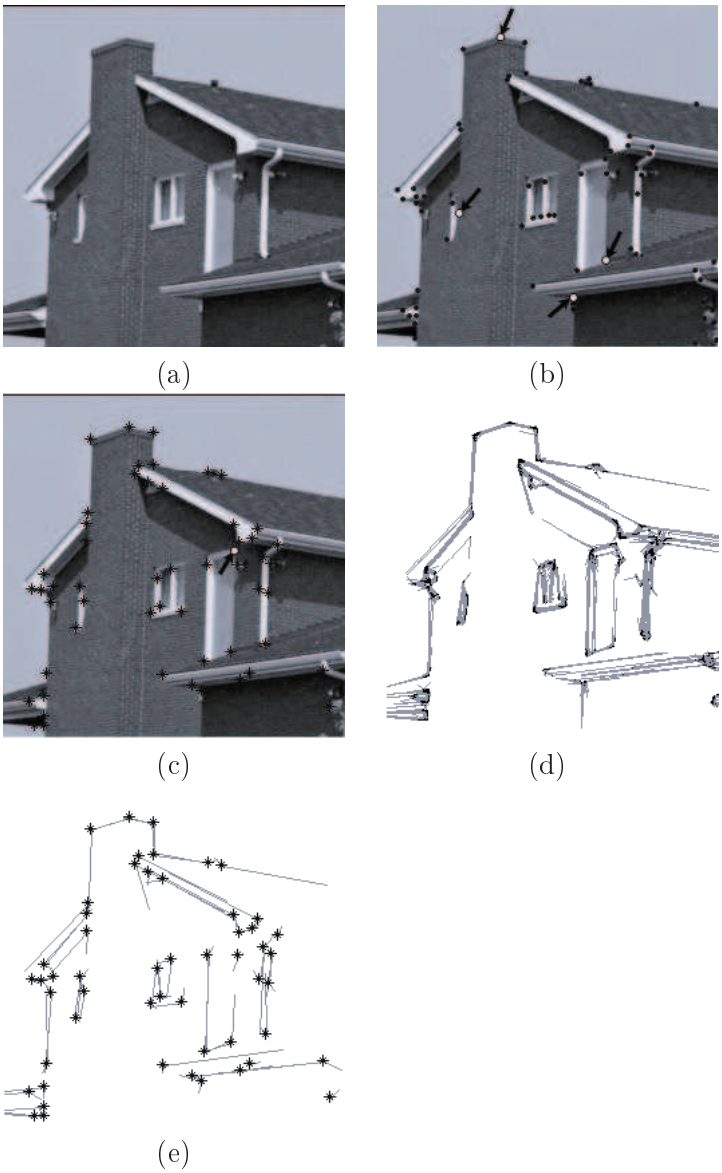


FIGURE 4.11 – (a) L'image "house". (b) Les jonctions obtenues par le détecteur de Harris. (c) Les jonctions obtenues par notre méthode avec $\mathcal{E} = 30$. (d) Les jonctions avant le filtrage avec $\mathcal{E} = 30$. (e) Les jonction après le filtrage avec $\mathcal{E} = 30$.

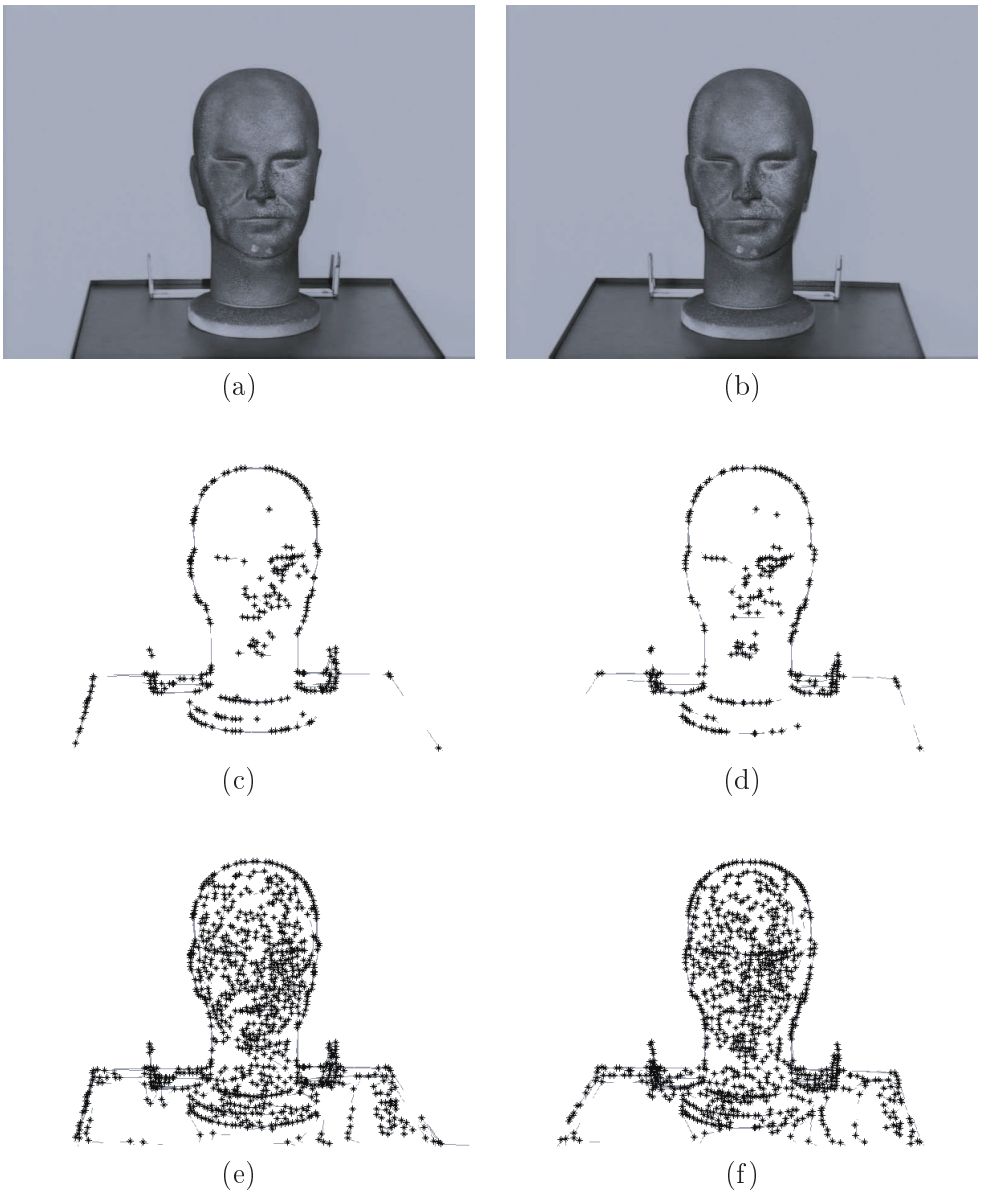


FIGURE 4.12 – Un couple d'images stéréo. (a) L'image gauche. (b) L'image droite. (c) Les jonctions de l'image gauche avec $\mathcal{E} = 20$. (d) Les jonctions de l'image droite avec $\mathcal{E} = 20$. (e) Les jonctions de l'image gauche avec $\mathcal{E} = 15$. (f) Les jonctions de l'image droite avec $\mathcal{E} = 15$.

4.5 Conclusion

Nous avons proposé des primitives image construites à partir des lignes de niveaux, adaptées chacune à une problématique donnée de l'analyse d'images, traitée dans le cadre de nos applications. Nous avons concentré nos efforts sur la définition d'une méthodologie cohérente dans laquelle un processus d'extraction de lignes de niveaux basique est enrichi afin de permettre la construction de primitives plus complexes. Par ailleurs, ces primitives ont d'une part suggéré la nature des processus de décision que nous avons défini par la suite (cumulatif par leur grand nombre). D'autre part, les descripteurs qui leur sont associés sont adaptés aux exigences et aux contraintes des dit-processus.

Au delà de cet exemple "ponctuel", nous nous sommes attachée, en détaillant le processus d'extraction, à montrer comment le choix d'une caractéristique adaptée, la ligne, et d'une procédure d'extraction adaptée, le suivi, conditionnent de manière naturelle un enchaînement logique de procédures élémentaires et de mesures associées (ex. fiabilité) renforçant la robustesse initiale dont elles tirent parti en préjugeant à leur tour des procédures d'appariement i.e. du niveau décision. Il s'agit donc bien, au stade de la description statique d'image, d'un premier exemple de **vision fruste**, c'est-à-dire d'exploitation d'un ensemble très restreint mais très cohérent d'objets et d'opérateurs au service de la perception minimale nécessaire à une classe d'actions.

Chapitre 5

Processus de décision cumulatif pour l'analyse d'images

5.1 Introduction

Les stratégies de décision cumulatives font l'objet de nombreuses études dans des domaines très divers : la résolution de problèmes en psychologie cognitive, la prise de décision (intelligence cumulative versus combinée) en finance et *management* ou encore en Intelligence Artificielle (accumulation d'évidences) en sont quelques illustrations. L'intérêt pour ces stratégies est tout aussi grand en fusion de données par accumulation de "décisions" issues des traitements sur différents capteurs qu'en analyse d'images où l'exemple le plus connu est sans doute la transformée de Hough dont on peut montrer l'analogie avec la transformée de Radon ou le filtrage adapté en traitement du signal [Mai85]. C'est à Paul Hough que nous devons l'idée initiale que des points alignés "partagent" des paramètres communs [1] (les paramètres de la droite les supportant) et qu'en changeant d'espace de représentation ces points sont "transformés" en droites concourantes. La transformée de Hough telle que nous la connaissons est le résultat de plusieurs contributions successives comme celle de Rosenfeld en 1969 qui écrit les équations algébriques associées et suggère l'utilisation d'un accumulateur puis celle de Duda et Hart qui ajoutent un formalisme issu de la Géométrie Intégrale pour représenter les droites et englober ainsi le cas particulier des droites verticales. A partir de là, de nombreuses variantes ont été proposées aussi bien pour détecter d'autres formes paramétriques (cercles, ellipses, etc.) que pour optimiser les calculs ou la mémoire requise pour l'accumulateur. C'est bien sûr ces différentes variantes qui nous ont inspirée lors de l'implémentation pratique de nos approches [2] :

- Pondération des votes pour tenir compte du degré de confiance accordé à certains points [OC73]. Le plus souvent la confiance est calculée à partir de l'amplitude du gradient.
- Quantification non régulière des cellules. En effet il est important de tenir compte du fait que les formes à détecter ne sont pas continues et que l'image comporte une dimension finie ce qui conduit à une inhomogénéïté des paramètres. Certains auteurs

1. donc votent "pour" des paramètres communs.
2. Nous ne citerons ici que les articles les plus anciens pour chacune des variantes mentionnées.

proposent alors une quantification au maximum d'entropie permettant d'assurer des comptes égaux pour chaque cellule de l'accumulateur [CT77].
- Vote à plusieurs tours [Ger87].
- Sélections des votants en utilisant l'orientation des gradients par exemple.
- Votes couplés. Cette stratégie appelée *Transformation m à 1* permet de tirer profit de connaissances locales pour faire correspondre plusieurs points à un seul point de l'espace de paramètres, ce qui permet un gain important en temps de calcul. Les connaissances locales les plus utilisées sont les dérivées partielles de la courbe en chaque point [Sha78].
- Incrémentation des accumulateurs de courbes voisines permettant de tenir compte du bruit et de l'incertitude sur la position des points "votants" [TD83].
- Stratégie adaptée de sélections des maximas dans l'espace de vote [GS84].

Dans les trois approches décrites dans ce chapitre, nous allons nous attarder systématiquement sur : la description des entités à cumuler et la stratégie de sélection de ces entités ; la définition de l'espace de cumul, ses paramètres et dimensions ; l'exploitation de cet espace pour établir une décision.

5.2 Décision cumulative binaire

Pour détecter les objets en mouvement à partir d'une caméra fixe, une approche répandue consiste à tenter de reconstituer la scène statique, appelée "fond", à partir d'une analyse statistique ponctuelle de la variation des niveaux de gris ou de tout autre descripteur(s) de primitives. Les objets en mouvement sont alors ceux dont les niveaux (ou descripteurs de primitives) ne concordent pas avec ceux de l'image de "référence". Cette image de référence [3], doit être constamment mise à jour. Les approches s'appuyant sur la construction d'une image de référence comportent donc deux étapes : la mise à jour à chaque nouvelle image acquise de l'image de référence et la comparaison de l'image courante avec l'image de référence afin d'établir la liste des objets mobiles. La principale difficulté de ces approches est de tenir compte des diverses perturbations pouvant affecter les descripteurs choisis. La mise à jour de la référence se doit d'être insensible aux fluctuations et bruits variés, aux variations d'illuminations pouvant affecter l'image, etc.

Afin de "reconstituer" la scène statique, les approches dites "naïves" consistent à déterminer le mode, la moyenne ou la médiane temporelle d'un descripteur de primitive sur une fenêtre temporelle donnée [Lv00], [CGPP03]. Ces techniques sont très coûteuses puisque une mémorisation des N images constituant la fenêtre temporelle est nécessaire. L'utilisation d'une moyenne exponentielle a ensuite été largement adoptée. Cette technique a été suggérée initialement par Charles C. Holt en 1957 dans le domaine de la prévision des séries temporelles [4]. Elle avait été proposée, sous la forme que nous lui connaissons, en 1963, par Brown [Bro63]. Cette approche appliquée à la construction des images de références souffre non seulement de n'être adaptée qu'au cas d'une distribution statistique

3. qui n'est pas forcément une image 2D au sens "collection de niveaux de gris : elle peut être un vecteur d'images si les descripteurs utilisés sont multi-dimensionnels par exemple.

4. Publiée à nouveau en 2004 pour une plus grand accessibilité [Hol04].

mono-modale mais aussi de ne pas fournir de méthodes explicites d'ajustement du seuil d'intégration à la référence. Ces dernières années, de nombreuses approches ont alors été proposées afin de tenir compte de la répartition statistique plus complexe des niveaux de gris et éventuellement de leur corrélation spatiale. Ainsi, dans [WADP97], les auteurs utilisent un modèle statistique multi-classes pour modéliser les objets en mouvement. Le modèle de construction de la référence est Gaussien pour chaque pixel. L'approche a été employée pour les scènes d'intérieur présentant peu de variations de contraste. [SG00a] proposent une approche dans laquelle la référence est modélisée par un mélange de Gaussiennes en chaque point (mixture of Gaussians). Dans cette approche, pour chaque point, le mélange de $K = 3, 5$ Gaussiennes est effectué. Les points qui ont un modèle statistique de variation non conforme à ceux de la référence sont considérés en mouvement. Les paramètres des Gaussiennes (moyenne et variances) et leur contribution dans le mélange sont les informations cumulées dans le temps. Le mélange de Gaussiennes est l'approche la plus répandue actuellement. Ces deux dernières approches ([WADP97] et [SG00b]) supposent une fonction densité de probabilité dont les paramètres sont obtenus durant une phase de modélisation préalable de la référence. Ces approches ne partent d'aucune hypothèse sur la distribution des données et estiment la fonction densité de probabilité afin de construire le modèle de la référence. La probabilité qu'un point n'appartienne pas à la référence est estimée grâce au modèle construit en utilisant les N précédentes images. Dans [EDHD02], les auteurs proposent une modélisation statistique de la référence basée sur l'estimation non paramétrique de la densité de certains noyaux, ce qui constitue une généralisation de la technique de mélanges de Gaussiennes où chaque échantillon parmi les N considérés est vu à son tour comme étant une distribution Gaussienne, cela permet d'estimer les fonctions densité de probabilité de manière plus précise en se basant sur les informations les plus récentes dans la séquence. Afin d'exploiter les corrélations spatiales, des approches sont fondées sur le principe du "mean shift" [HCD04] ou de la décomposition en valeurs singulières exploitant ainsi les matrices de co-variances [ORP00].

Nous constatons que la majorité des travaux basés sur la construction d'une image de référence ont majoritairement fait ce choix de considérer des primitives basiques (le point avec pour descripteur le niveau de gris ou la couleur) tout en concentrant tous les efforts sur la stratégie de sélection statistique des pixels "statiques", appartenant donc à la référence. Le lecteur aura remarqué que notre approche fait le pari inverse : nous avons concentré des effort significatifs sur la définition de primitives robustes vis-à-vis des diverses perturbations citées et nous relâchons nos efforts sur la procédure de sélection des primitives permanentes dans le temps. Notre choix des primitives robustes est tel que nous pouvons nous contenter d'un procédé simple de mise à jour par moyennage exponentiel par exemple, en adéquation totale avec l'approche intuitive qui consiste simplement à calculer l'occurrence d'un descripteur.

Dans cette section, nous décrivons de manière succincte notre procédé de mise à jour d'une référence basée sur les directions locales de lignes de niveaux. Pour plus de détails, on pourra se référer à [GBA00]. Nous mettons l'accent sur le procédé cumulatif en explicitant la nature des entités cumulées, la dimension de l'espace de cumul et le procédé de sélection des *maxima* de cet espace. Cette section sera illustrée par de nombreux résultats ayant pour principal objectif de montrer la pertinence du choix de primitives robustes malgré un processus de mise à jour de la référence très fruste.

5.2.1 Définition des entités à cumuler

On considère ici une extraction locale (sur un voisinage spatial donné) de groupes de lignes de niveaux droites passant par chaque point tel que décrit dans la section 4.2. Du point de vue de la primitive considérée (direction locale de lignes de niveaux), seule l'information "présence" ou "absence" d'une direction en un point donné est cumulée dans le temps. Les entités à cumuler sont alors binaires.

5.2.2 Description de l'espace cumulatif

L'espace de cumul est défini indépendamment en chaque point, il est de dimension η, où η est le nombre de directions de lignes de niveaux au maximum en chaque point (dépendant du degré de discrétisation). L'espace de cumul total est donc de dimension $M \times \eta$ pour une image de dimension M.

Introduisons quelques notations avant d'examiner plus en détail le processus de mise à jour de la référence.

Soient :

- $\theta_0, ..., \theta_\eta$: les η directions possibles de lignes de niveaux passant en un point.
- $f_t(p, \theta_k)$: la valeur prise à l'instant t par la direction associée au point p (avec k pouvant varier de 0 à $\eta - 1$). Cette fonction, à valeur dans $\{0, 1\}$, traduit simplement la présence ou l'absence d'une direction donnée au point p.
- $F_t(p, \theta_k) = \sum_{t=1}^{T} f_t(p, \theta_k)$: le nombre d'occurrences (fréquence d'apparition) de la direction θ_k au point p pendant la période d'observation T.

Le moyen le plus immédiat de déterminer si une orientation locale de lignes de niveaux passant par un point est suffisamment permanente pour appartenir à la scène statique est de calculer sa fréquence d'apparition dans une fenêtre glissante. La mise à jour de la référence consiste à réactualiser cette fréquence à chaque nouvelle image acquise.

$$F_i(p, \theta_k) = \sum_{i=1}^{t} f_i(p, \theta_k) = F_{i-1}(p, \theta_k) + f_t(p, \theta_k)$$

ou bien, sous la forme d'un filtre récursif du premier ordre :

$$F_i(p, \theta_k) = m \times F_{i-1}(p, \theta_k) + (1 - m) \times f_t(p, \theta_k)$$

avec : $m = \frac{t}{t+1}$

5.2.3 Décision

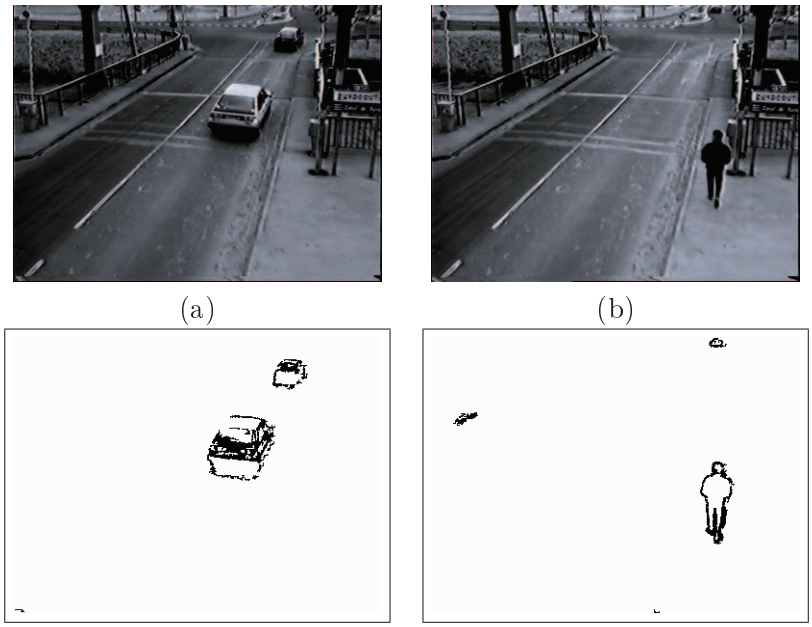
Une orientation dont la fréquence d'apparition est supérieure à un seuil T_0 sera considérée comme appartenant au fond et sera intégrée à la référence. Ce seuil peut être choisi de manière empirique, en fonction de la nature des objets en mouvement composant les scènes considérées. En effet, la permanence des directions des lignes de niveaux produites par le passage des objets en mouvement dépend au moins de quatre paramètres : la taille des objets, leur vitesse de déplacement, leur durée d'arrêt et la fréquence d'acquisition des

images. Nous pouvons partir de l'hypothèse qu'une direction est permanente si elle apparaît plus de C% de l'intervalle de temps considéré. Si $facq$ est la fréquence d'acquisition des images (facq images/seconde), et T l'intervalle de temps considéré, le nombre d'occurrences requis pour une direction donnée est alors : $T_0 = \frac{C}{100} facq$ Nous pouvons aussi, si la durée de présence d'un objet en mouvement sur l'image est connue (ou estimable par exemple à partir des données champs/vitesse moyenne) et égale à d secondes, en déduire le seuil nécessaire afin de ne pas intégrer les objets en mouvement dans la référence : $T_0 > d$

5.2.4 Résultats

Nous avons sélectionné ci-dessous les résultats permettant de rendre compte de la robustesse de l'approche, en particulier vis-à-vis des diverses perturbations telles que les changements de contraste. Par ailleurs, nous privilégions les résultats permettant une comparaison avec d'autres primitives classiques tels que les niveaux de gris bruts, les gradients ou les DoG. Afin que la comparaison ait un sens, le procédé de mise à jour de la référence et le seuil de permanence sont les mêmes quel que soit le type de primitives choisi.

FIGURE 5.1 – Les images (a), (b) et (c) sont extraites d'une séquences présentant des variations de contraste assez importantes. Les images (d), (e) et (f) donnent le résultat de la détection sans aucun filtrage additionnel. Seul le processus cumulatif ici sert à juger de la volatilité d'une direction de lignes de niveaux.



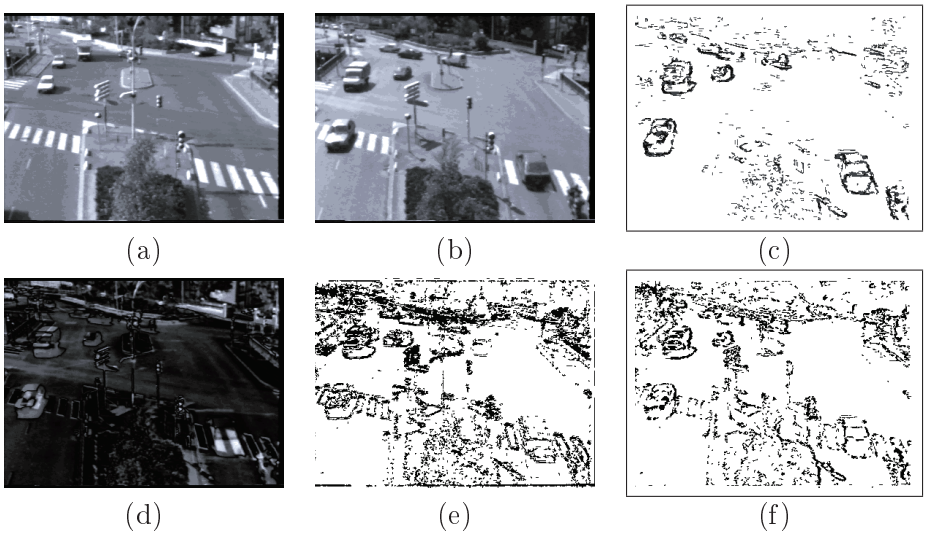
(a) (b)

(c) (d)

FIGURE 5.2 – Cette technique a été utilisée pour la détection de présence sur les ponts mobiles dans une collaboration entre la société CITILOG et VNF (Voies Navigables de France).

(a) (b)

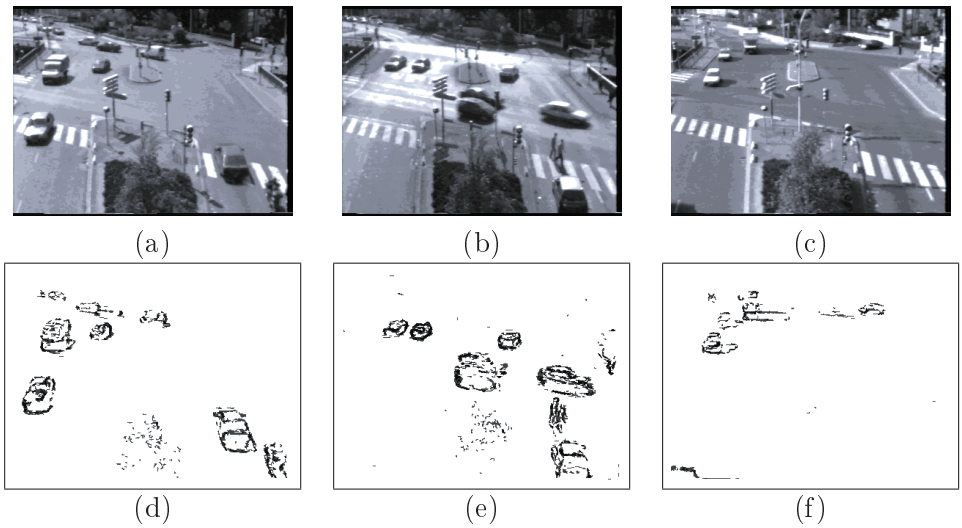
FIGURE 5.3 – Le cumul de direction de lignes de niveaux dans sa forme simplifiée a été employée à l'INRETS dans le cadre du projet Européen Cromatica (DG-XIII) pour la détection de stationnarités anormales dans le métro.



(a) (b) (c)

(d) (e) (f)

FIGURE 5.4 – (a) Image de référence. (b) Image courante prise 6 mois avant. (c) Image détection obtenue en utilisant un cumul de directions de lignes de niveaux. En plus des véhicules en mouvement, apparaissent les changements notamment dans la végétation ainsi que les ombres. (d) Détections obtenues avec cumul de primitives type "niveaux de gris". (e) Détections obtenues avec cumul de primitives type "DoG". (f) Détections obtenues avec cumul de primitives type "orientations de Gradient".



(a) (b) (c)

(d) (e) (f)

FIGURE 5.5 – Une image de référence d'un carrefour a été construite. Trois images extraites de la séquence sont analysées. (a) et (b) ont été enregistrées la même journée pluvieuse mais à des heures différentes. (c) a été enregistrée 6 mois plus tard un jour ensoleillé. (d), (e) et (f) représentent les images détection. Nous remarquons la robustesse et la stabilité vis-à-vis des variations de contraste.

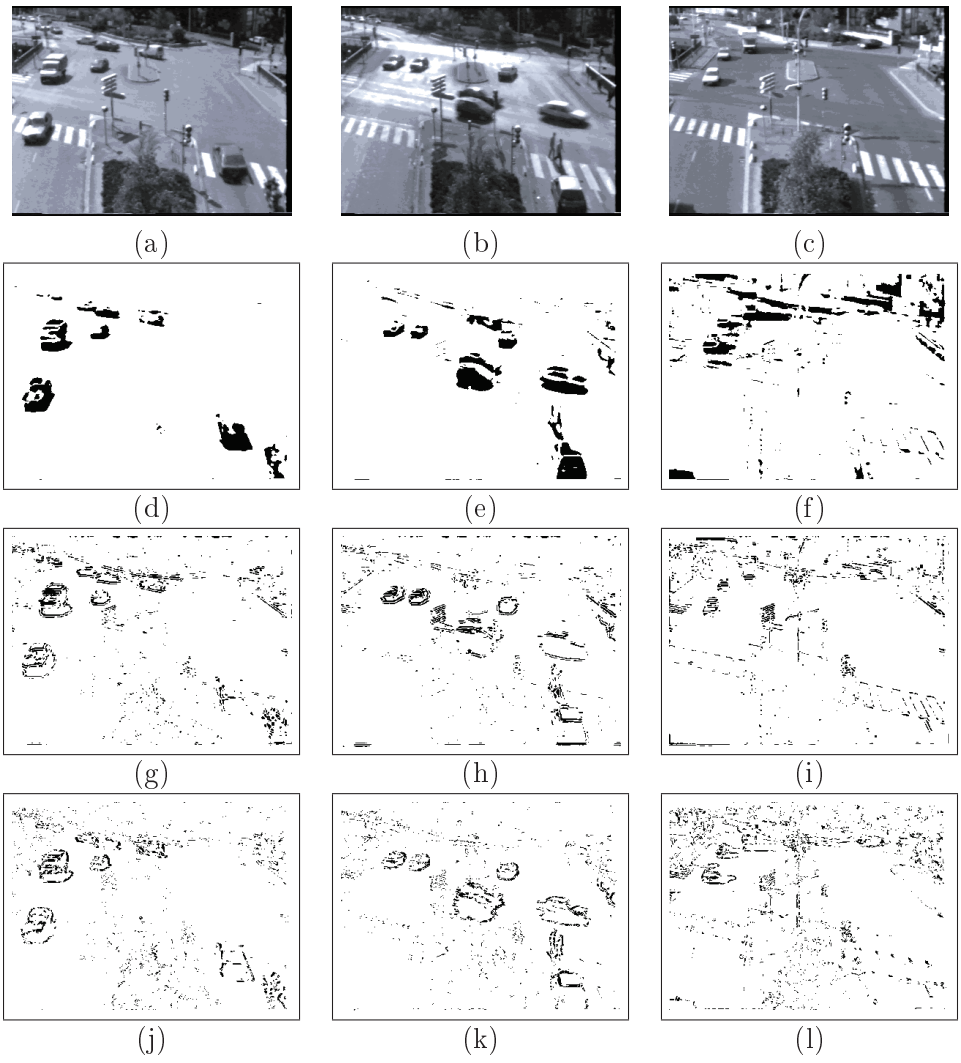


FIGURE 5.6 – Cette figure illustre les différents critères de détection pouvant être utilisés. (a), (b) et (c) sont les images extraites d'une séquence qui vont servir à la détection. Les images de chacune des lignes suivantes correspondent aux résultats de détection pour chacune des différentes primitives considérées. (d), (e), (f) sont les résultats correspondant à la primitive niveaux de gris. Le seuil de détection utilisé = 67 (seuil manuel nous permettant d'obtenir le meilleur résultat). (g), (h) et (i) correspondent aux résultats obtenus avec un Laplacien précédé par un lissage Gaussien (DoG). Le seuil de détection utilisé est ici très bas = 5 (mais cependant plus haut que celui que nous proposons pour les lignes de niveaux = 2 i.e. seuil de quantification+1), les détections sont donc plus sensibles aux faibles contrastes. (j), (k) et (l) représentent les résultats obtenus avec les directions de Gradient. Le seuil de détection est de 30 degré pour un seuil sur les amplitude de gradient de 30.

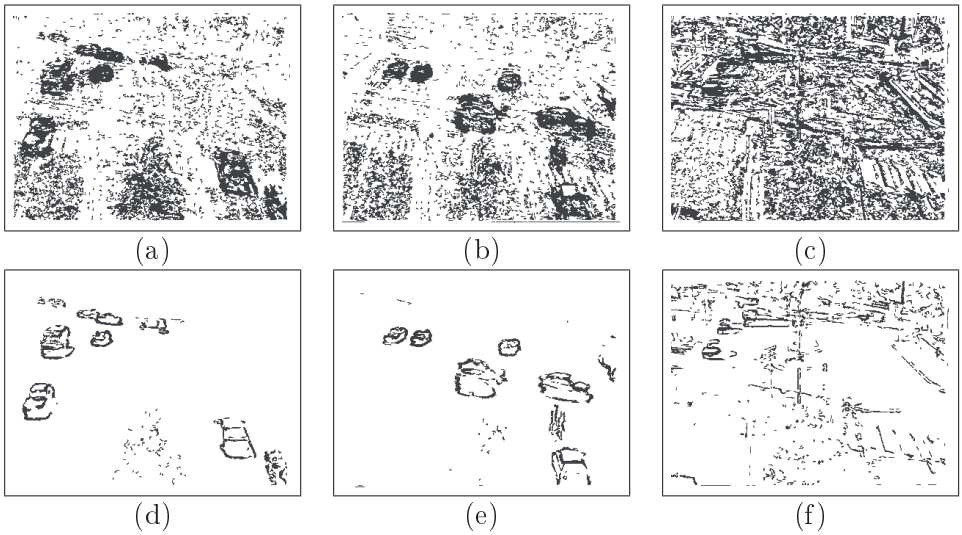


FIGURE 5.7 – Ces images montrent les résultats obtenus en utilisant un cumul de directions de gradient. La première ligne (images a, b et c) montre les résultats obtenus pour un seuil de permanence $T_0 = 30\%$ et un seuil de détection égal à 2 (différence d'amplitude de gradient minimum). Nous constatons que comparativement aux lignes de niveaux, les images de détection sont beaucoup plus bruitées. Dans les images (d), (e) et (f), le seuil de détection a été augmenté à 24. Nous obtenons alors des résultats similaires à notre approche mais le choix d'un seuil plus élevé empêche la détection des objets de faible contraste (voir images (d) et (e)). Enfin, nous constatons sur l'image (f) que l'augmentation du seuil n'a pas suffi a éliminer les fausses détections. Ce qui montre bien l'instabilité des directions de gradients par rapport aux lignes de niveaux.

5.2.5 Conclusion sur la décision cumulative binaire

Dans cette section, nous avons montré comment la robustesse des primitives choisies avait pour conséquence la possibilité de mettre en oeuvre un processus de décision cumulatif élémentaire voire binaire. Nous avons choisi délibérément un processus de mise à jour du fond basique, sans aucun filtrage (aucun lissage ou préfiltrage de l'image initiale, aucun filtrage des primitives, aucun filtrage des détections *a posteriori*). La qualité des résultats obtenus encourage la généralisation à la situation d'un capteur en mouvement, ce qui constitue l'objet des sections suivantes.

5.3 Décision cumulative multidimensionnelle

Nous étudions dans cette section le cas où la caméra est en mouvement. Nous considérons ici que le modèle de transformation est connu, cette dernière devant être estimée et les images recalées en conséquence. Nous nous limiterons aux cas où l'estimation de la profondeur des objets n'est pas nécessaire, ce qui limite le mouvement du capteur aux transformations suivantes : euclidienne, similarité, affine et projective (homographie). Ces dernières décennies ont vu apparaître une grande variété de techniques de mise en correspondance pour le recalage d'images, chacune adaptée à l'application visée. Le nombre croissant de ces nouvelles approches a eu pour conséquence la publication de synthèses très complètes permettant de comparer - selon des critères très divers - les techniques existantes [Bro92], [MV98], [LA99], [PJL+98], [Ana01]. Parmi les critères de distinction utilisés, nous citerons :

- La nature (fréquentielle ou spatiale) du domaine considéré. Les approches fréquentielles sont basées sur la corrélation des phases et exploitent des propriétés basiques de la transformée de Fourier (ou par ondelettes) [RC96]. Quant aux approches spatiales, elles s'appuient sur une extraction et une comparaison de primitives image.
- La nature de la transformation recherchée (euclidienne, affine, projective ou élastique). Cette transformation dépend en général de l'application considérée. Chaque transformation particulière implique des invariants adéquats, calculés à partir des primitives extraites : longueur, orientation, rapports de longueur, birapport, etc.
- Les primitives choisies. Les plus usuelles sont les points particuliers [Mor81], [Low04], [BTG06], les contours [Nac77], [MN84], les surfaces [PCSW89], les régions [GSP86], les lignes [SKB82], ou les descripteurs de Fourier [KG82].
- La mesure de similarité entre primitives, qui dépend aussi du type de primitives utilisées. Les plus classiques sont : la corrélation croisée [RK82], la somme des différences en valeur absolue [BS72] ou la distance de Levenstein [Gui86].
- L'espace de recherche et la stratégie d'appariement. De manière simplifiée, les stratégies sont basées sur une mise en correspondance de primitives qui est explicite (ex. recherche exhaustive, programmation dynamique [Gui86], relaxation [SH90], transformée de Hough [Bal81]), ou implicite (ex. erreur quadratique, minimisation [Sze94], programmation linéaire [Bai85]).

Comme nous l'avons évoqué en introduction de cette partie, notre étude est partie de trois constatations :

- D'abord, les images à recaler sont rarement acquises dans les mêmes conditions d'éclairement. Pourtant, la plupart des primitives utilisées dans la littérature sont sensibles aux variations de contraste. Elles sont soit directement dépendantes des niveaux de gris, ou alors souffrent d'un manque de techniques réellement indépendantes du contraste (et/ou nécessitent des seuils difficiles à ajuster) capables de les extraire [CCM99].
- Ensuite, les motifs répétitifs - pourtant fréquents dans les images naturelles - engendrent des ambiguïtés d'appariement qui sont rarement prises en considération dans les stratégies de mise en correspondance existantes.
- Enfin, dans les applications temps-réel, la robustesse du système complet tirerait profit d'un processus de décision progressif qui facilite un contrôle rapide par une

première estimation des paramètres de la transformation recherchée, puis si nécessaire son raffinement.

Notre approche tente alors de fournir des éléments de réponse aux problèmes cités ci-dessus. Pour cela, nous avons choisi une classe de primitives adaptées : les lignes de niveaux, robustes vis-à-vis de variations de contraste [CCM99], [MG00] [5]. Nous avons utilisé les lignes de niveaux avec succès en analyse du mouvement, en particulier pour des scènes d'extérieur, là où l'hypothèse d'invariance de la luminosité n'est pas vérifiée [GBA00], [Bou98]. De plus, elles sont particulièrement adaptées à une stratégie d'appariement basée sur un processus de vote. En effet, le nombre important de lignes de niveaux dans l'image y entraîne de fortes redondances locales et permet d'envisager un processus de mise en correspondance cumulatif où chaque ligne est appelée au vote à une phase donnée en fonction de sa fiabilité. Pour restreindre l'espace de vote, nous exploitons une technique de décision sur graphe bi-partite dite des mariages stables [GI89], [KMV94]. Chaque primitive crée une liste de préférence des primitives dans l'autre image, classées des plus aux moins similaires en fonction de mesures ou métriques adaptées. Ces préférences pondèrent le processus de décision qui se déroule en plusieurs étapes : les primitives les plus fiables participent d'abord au vote, les autres sont sollicitées ensuite pour confirmer ou infirmer le vote précédent. La transformation retenue, issue des couples de primitives, est celle qui recueille le maximum de votes.

Plus précisément, la notion de fiabilité introduite dans le chapitre précédent nous amène à envisager une stratégie de vote à plusieurs tours où chaque phase permet à une nouvelle catégorie de votants d'exprimer leur opinion. La technique des mariages stables [GI89], [KMV94], [ZSZ00] suggère de construire pour chaque primitive une liste de préférence des $\kappa_{\max}$-plus proches primitives trouvées dans l'autre image. Le *couple* inter images [primitive,correspondant potentiel] permet alors d'obtenir une estimée et donc un vote pour une transformation donnée. Ce vote aura plus ou moins de crédit en fonction de la fiabilité des primitives votant et de la position du candidat dans la liste de préférence.

5.3.1 Définition des entités à cumuler

Nous considérons que la phase de construction des primitives par groupement de segment de lignes de niveaux a été réalisée, nous fournissant ainsi un ensemble $C^k_{I\,k=1,N}$ de vecteurs descripteurs de primitives pour la première image I et un ensemble $C^k_{J k=1,M}$ de vecteurs descripteurs de primitives de l'image J. Rappelons que les descripteurs sont : les invariants calculés ainsi que les contrastes moyens de part et d'autre des segments formant la primitive. Les coordonnées du point milieu de chaque segment formant la primitive seront stockés dans le vecteur caractéristique dans le but d'établir le système d'équation à résoudre pour estimer les paramètres de la transformation. Les primitives ainsi constituées vont s'associer à d'autres primitives de l'autre image formant ainsi des **couples** de primitives. En se plaçant toujours du point de vue de la primitive image extraite, celle-ci génère une liste de préférence de primitives de l'image suivante construisant ainsi des "hypothèses" de transformations inter-images. Ces hypothèses vont ou non être confirmées par le vote

5. Soulignons que d'autres existent : Li et al. [LTD00] exploitent par exemple une variable de chromaticité par nature indépendante de la luminosité

des autres primitives images liées par la même transformation. La décision cumulative n'est plus binaire puisqu'il ne suffit plus de vérifier la seule présence de la primitive dans l'image suivante, il faut rechercher sa nouvelle position, induisant ainsi une hypothèse de transformation. Toutes les primitives liées par la même transformation participent à l'émergence de la solution recherchée, i.e. l'estimation du modèle de mouvement. Par ailleurs, en fonction de la fiabilité des primitives (voir section 4.2), celles-ci peuvent être classées en plusieurs catégories, entrainant un processus de décision à plusieurs phases.

Construction des listes de préférence à longueur variable : Nous inspirant de la technique des mariages stables, chaque primitive $u \in \left\{\overrightarrow{C_I^k}\right\}_{k=1,N}$ de l'image I construit une liste de préférence triée de primitives parmi les primitives de l'image J, des plus similaires aux moins similaires, à l'aide d'une mesure de distance euclidienne sur les descripteurs de la primitive. A titre d'exemple, pour des primitives de type "Angle", dont les descripteurs sont le rapport des longueurs, la différence des orientations et le contraste moyen de part et d'autres des lignes, la distance *dist* entre u et $v \in \left\{\overrightarrow{C_J^k}\right\}_{k=1,M}$ est définie simplement par :

$$dist(u,v) = k_1 \left|\Delta\theta_{\mathbf{u}} - \Delta\theta_{\mathbf{v}}\right| + k_2 \left|\ell_{\mathbf{u}} - \ell_{\mathbf{v}}\right| + k_3 \left(\left|c_{i\mathbf{u}} - c_{i\mathbf{v}}\right| + \left|c_{j\mathbf{u}} - c_{j\mathbf{v}}\right|\right).$$

k_1, k_2 et k_3 permettent d' ajuster l'importance relative de chaque caractéristique. Il est à noter que $\sum k_i = 1$ et $k_3 < k_{i=1,2}$ car le contraste n'est pas une caractéristique invariante mais reste néanmoins à considérer avec un plus faible poids. Bien sûr, chaque différence en valeur absolue est normalisée par la valeur maximale prise afin d'obtenir une distance finale comprise entre 0 et 1 (de la plus à la moins ressemblante).

Le procédé est identique pour les autres types de primitives. Seul le nombre d'invariants et de segments constituant la primitive change.

Chaque primitive construit ainsi sa liste des $\kappa-$plus proches ($\kappa < \kappa_{\max}$) primitives en utilisant la distance définie ci-dessus. Il est important de noter que la taille de la liste de préférence est variable d'une primitive à l'autre. Les nouveaux candidats sont insérés dans cette liste jusqu'à ce que la distance obtenue devienne trop élevée. En fait, le choix de listes de préférences variables est nécessaire afin d'éviter l'élimination arbitraire d'un candidat en raison de tailles de listes fixées *a priori* s'avérant inadaptées *a posteriori*. Dans la pratique, nous avons fixé prophylactiquement $\kappa_{\max} = 10$, qui n'a jamais été atteint. Adapter cette borne automatiquement, si cela s'avérait nécessaire, en fonction de la distribution des *segments* (longueurs, orientations), ne présente pas de difficulté.

Classement des primitives en catégories : Les primitives sont classées en plusieurs catégories pour préparer la décision. La FIGURE 5.8 illustre le découpage en 3 catégories en fonction de la longueur et du contraste moyen de part et d'autre des segments constituant chaque primitives. Plus précisément, nous considérons les moyennes et écart types des longueurs ($\overline{l}$, σ_l) et contraste ($\overline{c}$, σ_c) sur toute l'image. Les primitives les plus fiables sont celles correspondant aux longueurs et contrastes les plus élevés, i.e. plus grand que $\overline{l} + \sigma_l$ et $\overline{c} + \sigma_c$ respectivement. La seconde catégorie est attribuée à celles dont les longueurs sont élevées mais le contraste plus faible, i.e. ($l > \overline{l} + \sigma_l$ et $\overline{c} < c < \overline{c} + \sigma_c$). Enfin, la dernière

catégorie, regroupe les primitives de faible longueur et faible contraste avec : ($\bar{l} < l < \bar{l} + \sigma_l$ et $c > \bar{c} + \sigma_c$). Les primitives restantes sont écartées.

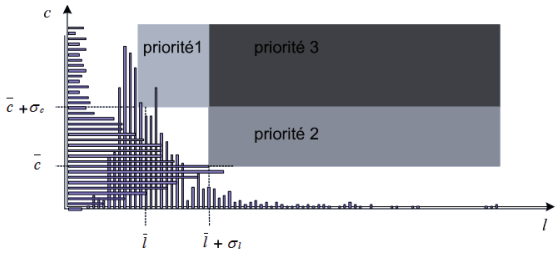


FIGURE 5.8 – Classement des primitives en plusieurs catégories.

5.3.2 Description de l'espace cumulatif

L'espace de vote V est multidimensionnel : à 4 dimensions pour une transformation de similarité (dx, dy, α, ψ) (2 translations, 1 rotation et 1 échelle), à 6 dimensions pour une transformation affine $(a_{11}, a_{12}, a_{21}, a_{22}, t_x, t_y)$ et à 8 pour une transformation projective $(h_{11}, h_{12}, h_{13}, h_{21}, h_{22}, h_{23}, h_{31}, h_{32})$. Chaque couple de primitives (u, v) vote - si le couple se préfère mutuellement [6] - en fonction de la position de v dans la liste de préférence de u pour une transformation, celle estimée par résolution d'un système d'équations dans lequel les coordonnées du point milieu des segments constituant les primitives du couple apparié apparaissent. La résolution du système peut être soit directe lorsque celui-ci est constitué de peu d'équations soit effectuée à l'aide d'une méthode de résolution aux moindres carrés.

Cas de la transformation de similarité : Le système obtenu pour un couple de primitives et 4 points (chacun fournissant 2 équations) est résolu directement de manière analytique.

$$\begin{pmatrix} x'_1 \\ y'_1 \\ x'_2 \\ y'_2 \end{pmatrix} = \begin{pmatrix} x_1 & y_1 & 1 & 0 \\ y_1 & x_1 & 0 & 1 \\ x_2 & -y_2 & 1 & 0 \\ y_2 & x_2 & 0 & 1 \end{pmatrix} \begin{pmatrix} \psi\cos(\alpha) \\ \psi\sin(\alpha) \\ dx \\ dy \end{pmatrix}$$

Cas de la transformation affine : L'objectif est de trouver les vecteurs $\mathbf{h}$, $\mathbf{k}$ qui minimisent les normes $\left\|\mathbf{x}' - M\mathbf{h}\right\|$ et $\left\|\mathbf{y}' - M\mathbf{k}\right\|$.

$$\begin{pmatrix} x'_1 \\ x'_2 \\ x'_3 \\ x'_4 \end{pmatrix} = \begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \\ x_4 & y_4 & 1 \end{bmatrix} \begin{pmatrix} a_{11} \\ a_{12} \\ t_x \end{pmatrix} = M\mathbf{h} \quad \begin{pmatrix} y'_1 \\ y'_2 \\ y'_3 \\ y'_4 \end{pmatrix} = \begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \\ x_4 & y_4 & 1 \end{bmatrix} \begin{pmatrix} a_{21} \\ a_{22} \\ t_y \end{pmatrix} = M\mathbf{k} \tag{5.1}$$

6. Préférence mutuelle = "u est dans la liste de préférence de v **et** v est dans la liste de préférence de u".

La matrice H_A est estimée en résolvant le système (5.1) par moindre carrés :

$$\mathbf{h}(resp.\mathbf{k}) = (M^T M)^{-1} M^T \mathbf{x})(resp.\mathbf{y}) \tag{5.2}$$

Cas de la transformation projective : Chaque primitive fournit 4 points, chaque point générant 2 équations, le système d'équations suivant qui en découle est résolu aux moindres carrés.

$$\begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -x_1\acute{x}_1 & -y_1\acute{x}_1 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -x_1\acute{y}_1 & -y_1\acute{y}_1 \\ x_2 & y_2 & 1 & 0 & 0 & 0 & -x_2\acute{x}_2 & -y_2\acute{x}_2 \\ 0 & 0 & 0 & x_2 & y_2 & 1 & -x_2\acute{y}_2 & -y_2\acute{y}_2 \\ x_3 & y_3 & 1 & 0 & 0 & 0 & -x_3\acute{x}_3 & -y_3\acute{x}_3 \\ 0 & 0 & 0 & x_3 & y_3 & 1 & -x_3\acute{y}_3 & -y_3\acute{y}_3 \\ x_4 & y_4 & 1 & 0 & 0 & 0 & -x_4\acute{x}_4 & -y_4\acute{x}_4 \\ 0 & 0 & 0 & x_4 & y_4 & 1 & -x_4\acute{y}_4 & -y_4\acute{y}_4 \end{bmatrix}^{-1} \begin{bmatrix} x'_1 \\ y'_1 \\ x'_2 \\ y'_2 \\ x'_3 \\ y'_3 \\ x'_4 \\ y'_4 \end{bmatrix} = \begin{bmatrix} h_{11} \\ h_{12} \\ h_{13} \\ h_{21} \\ h_{22} \\ h_{23} \\ h_{31} \\ h_{32} \end{bmatrix} \tag{5.3}$$

Quel que soit le type de transformation considéré, la contribution $\triangle V$ d'un vote pour une transformation donnée entre 2 primitives u et v est calculée en tenant compte des considérations ci-dessous. Soit $\triangle V = a_1 \times a_2 \times a_3 \times a_4$, où les paramètres a_1, a_2, a_3, a_4 sont ajustés comme suit :

1. Un correspondant potentiel v_{pos} aura une plus grande contribution s'il se trouve en début de liste de préférence et symétriquement en u. Le coefficient a_1 est alors inversement proportionnel à une fonction des positions pos_v (resp. pos_u) de la primitive v_{pos_v} (resp. u_{pos_u}). Choisissons [7] :
$$a_1 = \frac{\mathbf{1}}{\sqrt{pos_v \times pos_u}} \;\; ; a_1 \in \left\{\tfrac{1}{\kappa}, ..., 1\right\}$$
$$/ \sup(\, pos_u, pos_v) \leq \kappa < \kappa_{\max}$$
2. A un tour de vote donné, les primitives de catégorie inférieure ou égale à t ($t < W_{\max}$ est l'itération courante du processus de vote) sont autorisées à s'associer à d'autres primitives de la seconde image. Au tour $i, t = i$ et les votants doivent avoir une priorité au moins de i. Les primitives de priorité supérieure vont voter avec plus de poids. a_2 est alors proportionnel à la fiabilité de la primitive :
$$a_2 = \tfrac{1}{w_{\mathbf{u}}};\; a_2 = \left\{\tfrac{1}{W_{\max}}, ..., 1\right\}$$
3. Les primitives les plus ressemblantes (distance faible) votent avec un poids plus grand :
$$a_3 = (1 - dist(\mathbf{u}, \mathbf{v})) \; ; a_3 \in [0...1]$$
4. Enfin, puisque tous les a_i $i = 1, 3$ appartiennent à l'intervalle $[0, 1]$, nous pouvons utiliser un facteur d'échelle a_4 afin d'obtenir des incréments entiers.

7. D'autres fonctions symétriques ou non de u et v peuvent être testées en regard de l'application.

5.3.3 Décision

L'estimation de la transformation recherchée est réalisée à travers un processus de vote à plusieurs tours (l'approche proposée est schématisée dans la FIGURE 5.9). A chaque tour de vote, une nouvelle catégorie de *couples* exprime son opinion (une estimée de la transformation recherchée) en fonction de sa fiabilité. Cette stratégie a deux avantages :

1. La disponibilité à chaque tour d'une approximation même grossière de la transformation recherchée peut être - même si elle n'est pas très précise - exploitée dans certaines applications nécessitant un résultat immédiat.
2. En raison des motifs répétitifs présents dans les images naturelles, conduisant fatalement à des ambiguïtés d'appariement, le processus qui consiste à inviter au vote des primitives moins fiables - mais souvent plus nombreuses - donne une nouvelle chance à l'émergence de la solution exacte.

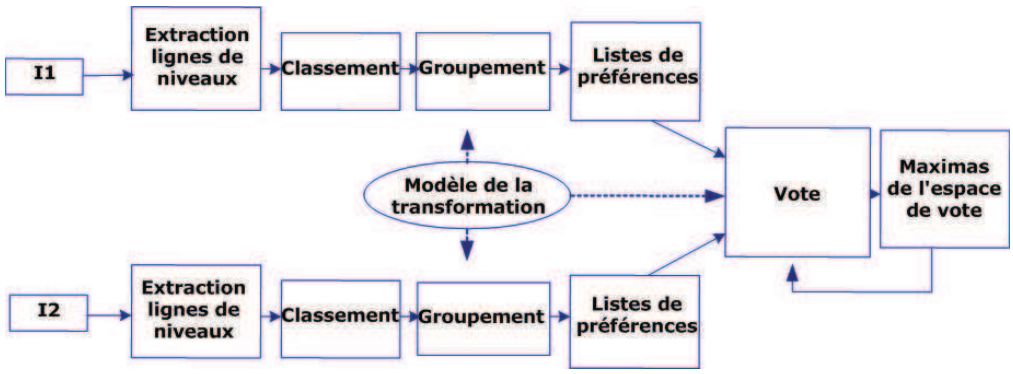


FIGURE 5.9 – Schéma général du processus de vote.

Les *couples* sont classés en autant de catégories que de tours de votes en divisant simplement la liste de primitives. Au premier tour, la première catégorie vote. Si un pic unique apparaît dans l'espace de vote alors une transformation obtient une majorité absolue : le processus de vote peut donc s'arrêter. Sinon, un second tour peut démarrer autorisant une nouvelle population à voter : les primitives ayant une fiabilité moindre. S pics dans l'espace de vote sont sélectionnés. Les valeurs cumulatives des *maxima* sélectionnés sont amplifiées pour le tour suivant afin de privilégier les décisions des primitives les plus fiables. Le vote s'arrête lorsqu'il n'y a plus de primitives ou lorsqu'un pic émerge, récoltant ainsi la majorité absolue.

5.3.4 Résultats

Transformation de similarité : Dans le cadre d'un travail sur le recalage d'images microscopiques, nous avons comparé notre approche avec 2 techniques classiques : la première est basée sur une corrélation de phases (utilisant la TFR), la seconde sur une minimisation de distance entre intensités (méthode d'optimisation de Levenberg-Marquardt). Aucune n'a donné les résultats escomptés en particulier sur les images contenant de nombreuses répétitions. Nous avons aussi testé une démonstration web en ligne basée sur une approche

différentielle [FFKM02]. Là encore, la transformation obtenue n'est pas correcte, voir FIGURE 5.10 (f). Nous avons sélectionné ci-dessous des résultats obtenus avec des images particulièrement difficiles à traiter en raison des variations de contraste et des nombreuse répétitions créant des ambiguïtés (voir FIGURES 5.10, 5.11 et 5.12).

Transformation Affine : Les FIGURES 5.13 et 5.14 montrent quelques résultats obtenus lorsque l'hypothèse d'un mouvement affine était choisie. L'existence d'un plan perpendiculaire à l'axe optique, majoritaire dans l'image, rend cette hypothèse de transformation quasi valide dans les cas choisis pour cet exemple. Le recalage puis la superposition des images montrent l'erreur de recalage plus importante sur les zones ne correspondant pas à ce plan servant de référence au recalage.

Transformation projective : La figure FIGURE 5.15 donne quelques résultats dans le cas d'une transformation projective. L'existence d'un plan majoritaire dans l'image (bâtiment) a pour conséquence, à travers le choix d'un pic dans l'espace de vote, une estimation de l'homographie associée à ce plan de référence.

Rôle du découpage en catégories : Afin de montrer le rôle dans l'estimation de la transformation finale des différentes catégories de primitives, nous calculons ici l'erreur de recalage pour chaque point, catégorie par catégorie. Les zones dans l'image dont l'erreur est importante correspondent *a priori* à un autre modèle de transformation que celui qui a été pré-établi (FIGURE 5.16).

5.3.5 Conclusion sur la décision cumulative multidimensionnelle

L'approche proposée pour recaler des images partant d'un *a priori* sur le modèle de la transformation recherchée a été conçue au départ pour faciliter la réutilisabilité des primitives construites dans l'éventualité où le modèle de transformation se complexifierait. C'est dans cet esprit que nous comptons étendre cette approche à la prise en compte graduelle des transformations de la moins à la plus complexe. Il serait envisageable de mettre en place un processus multi-phases, permettant d'estimer d'abord les transformations de similarité (modèle le plus simple) associées à des zones différentes de l'image à travers l'extraction de plusieurs pics dans l'espace de vote. Les zones n'ayant pas voté pour les pics choisis sont alors probablement déformées par un autre modèle plus complexe : affine, puis en utilisant le même procédé, projectif. La phase suivante consiste alors à enrichir les primitives construites pour qu'elles soient adaptées au nouveau modèle à estimer. Et ainsi de suite. Cette extension a plusieurs avantages : d'abord l'estimation de plusieurs transformations pour un même modèle mais associées à différentes zones de l'image. Ensuite, l'estimation de plusieurs modèles de transformations pouvant cohabiter au sein d'une même image. Pour résumer : dans le cas d'une même transformation globale liant les primitives, le multi-tour (au niveau du vote) permet de fournir la transformation majoritaire (le plan qui "saute au yeux") puis les suivantes par ordre d'importance visuelle, le multi-phase permettra de gérer plusieurs modèle de transformations au sein d'une même image.

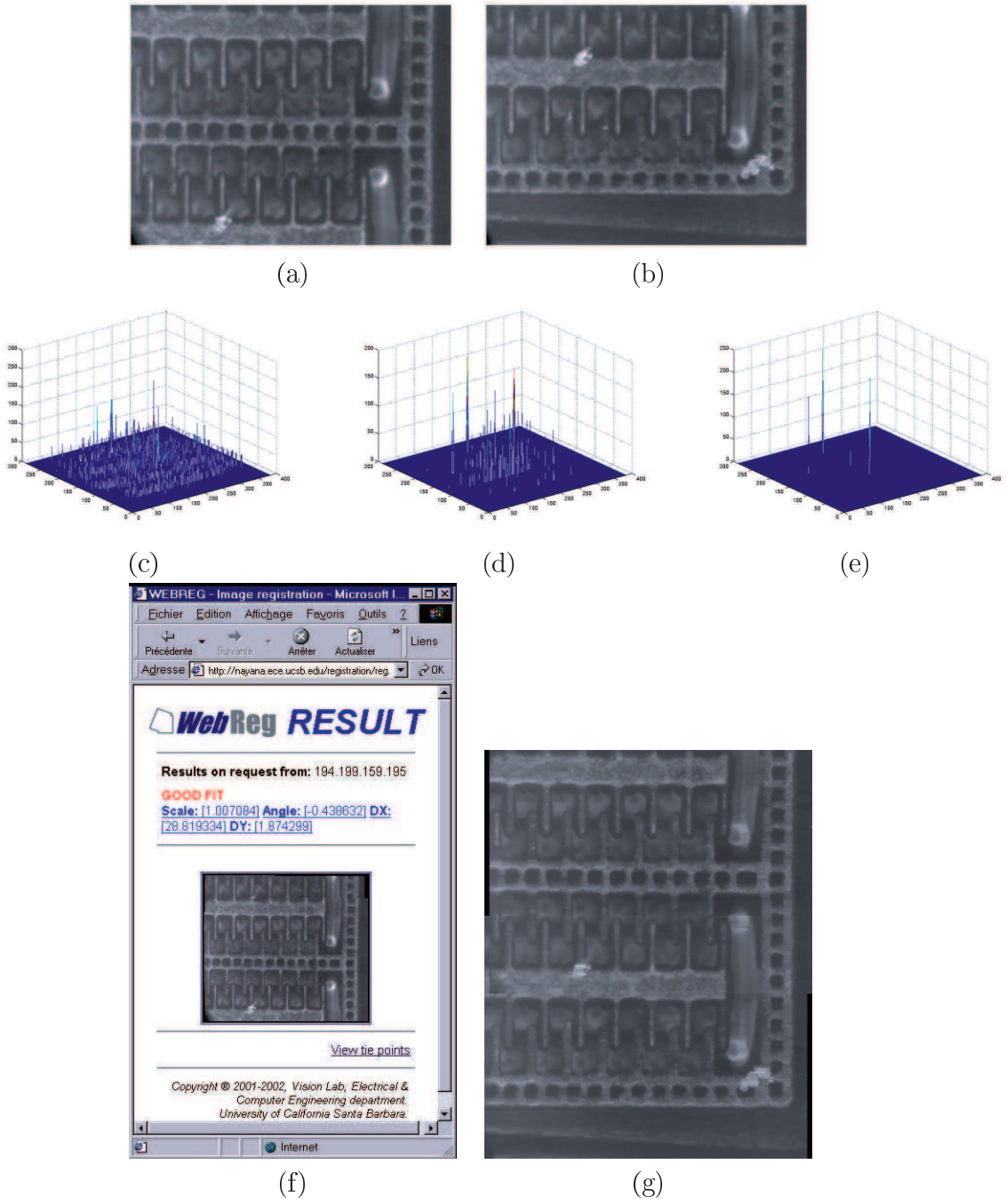


FIGURE 5.10 – (a), (b) Images obtenues par un microscope électronique à balayage de franges qui correspondent à des profils partiels d'un peigne électrostatique en cuivre d'un micro gyromètre. (c) à (e) Les espaces de vote en translation pour 3 tours de vote. (f) Résultats obtenus en utilisant une approche classique différentielle basée sur une minimisation de distance d'intensités (utilisant la méthode Levenberg-Marquardt pour l'optimisation). En raison de la présence de motifs répétitifs dans les images, la transformation obtenue n'est pas correcte. (g) La transformation correcte estimée (translation de -98, 3, rotation = 0, échelle = 1) malgré les motifs répétitifs. Ce résultat est très satisfaisant pour le recalage d'images de micro-dispositifs obtenues par un microscope électronique dont le champ est limité.

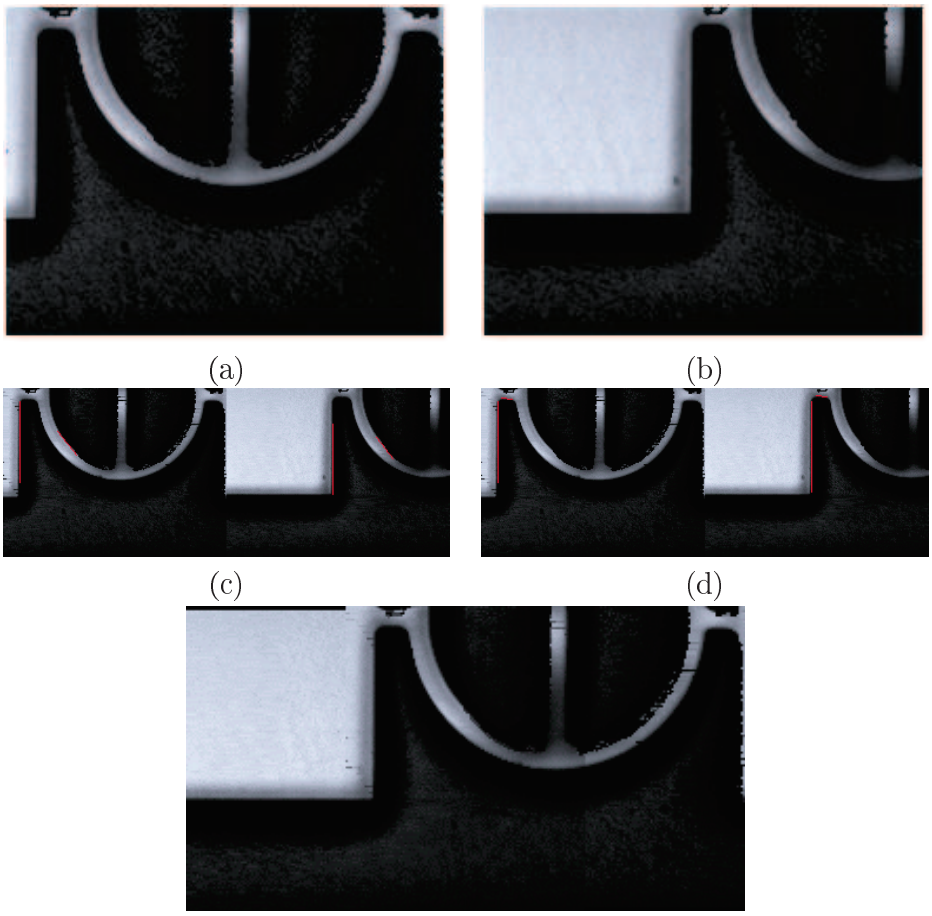


FIGURE 5.11 – (a) et (b) Images à recaler prises par un microscope électronique. Elles correspondent aux 2 profils partiels d'un anneau en aluminium. (c) et (d) Exemple de couples se préférant mutuellement. La dynamique des images a été réduite de 2 afin de mettre en évidence les couples choisis en rouge. (e) Raccord correct obtenu : ici une translation de (-2, 78) entre les deux images.

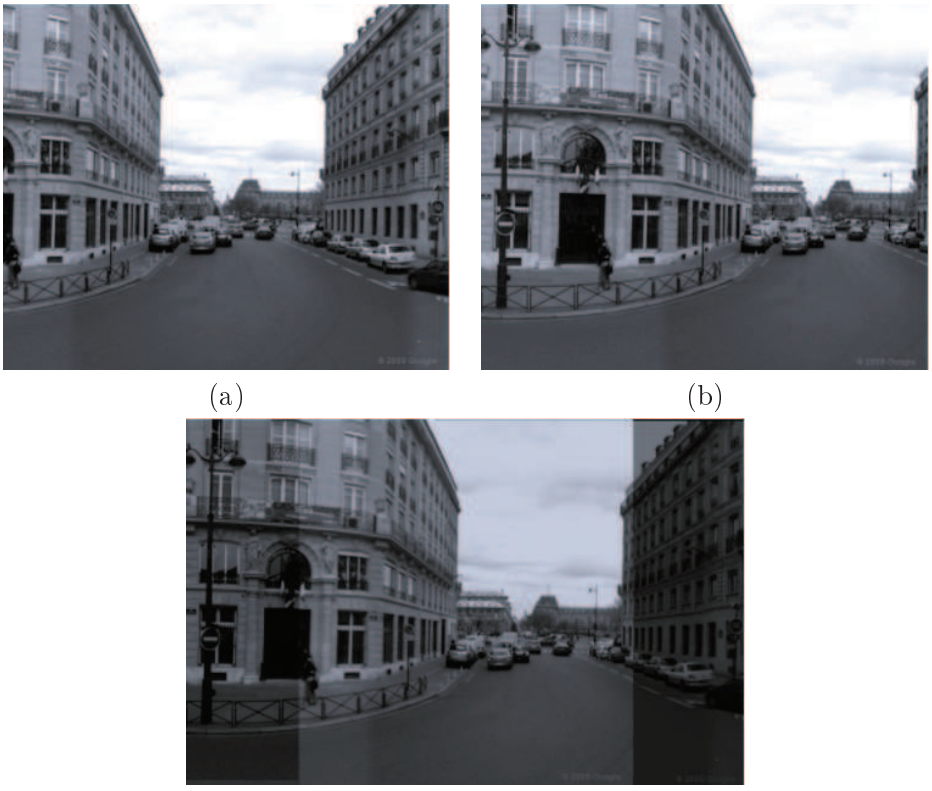
(a) (b) (c)

FIGURE 5.12 – (a) image 1. (b) image 2. (c) Raccord obtenu : la dynamique de l'image 1 a été réduite pour transparence. Après transformation, elle est superposée sur l'image 2. La transformation trouvée et effectuée est : dx = 5, dy = 3, teta = 9, échelle = 1, 0096. Le recalage est satisfaisant pour les besoins de commande dans ce type d'applications.

(a) (b)

(c)

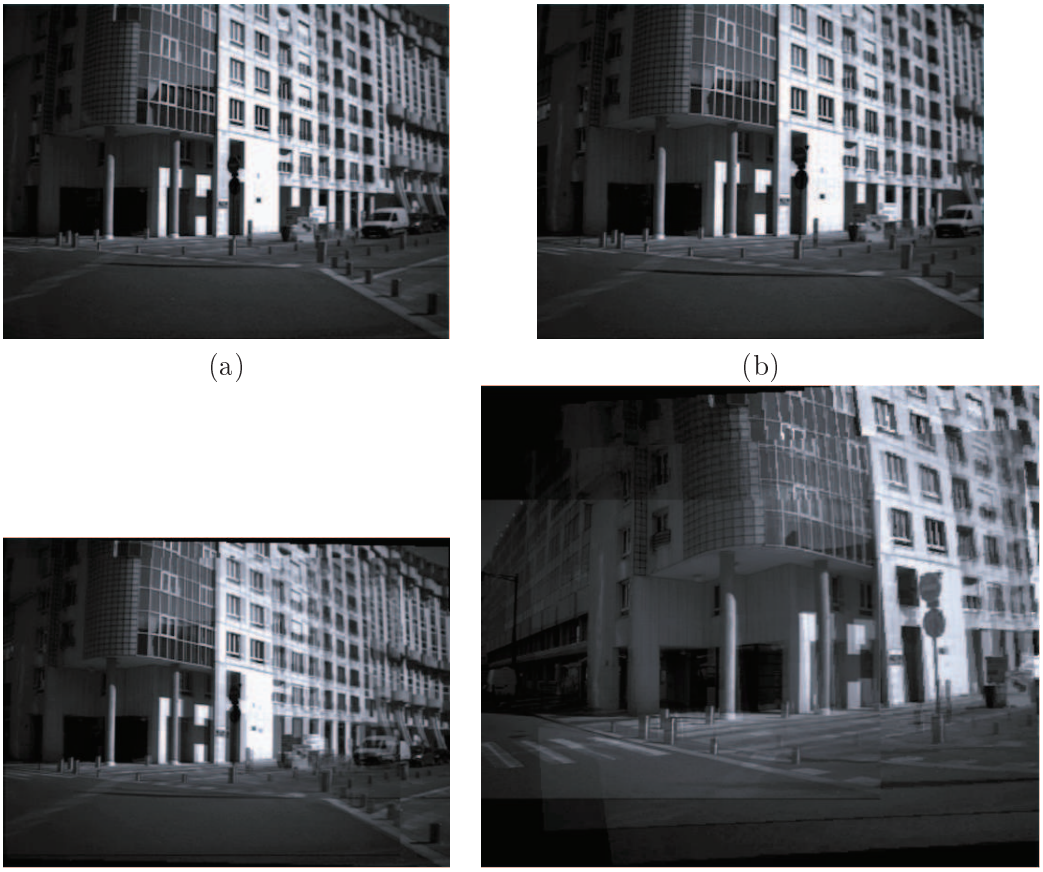
FIGURE 5.13 – (a) et (b) Deux images successives d'une séquence. (c) Résultat de l'alignement en supposant un modèle de transformation affine.



(a) (b)

(c)

FIGURE 5.14 – (c) Résultat de l'alignement des images (a) and (b) dans le cas d'un modèle de transformation affine



(a) (b)

(c) (d)

FIGURE 5.15 – (c) Alignement des deux images (a) and (b) obtenues à partir d'une caméra en mouvement. La scène est essentiellement composée de plans, ce qui rend l'hypothèse de transformation projective valide. (d) Alignements multiples.

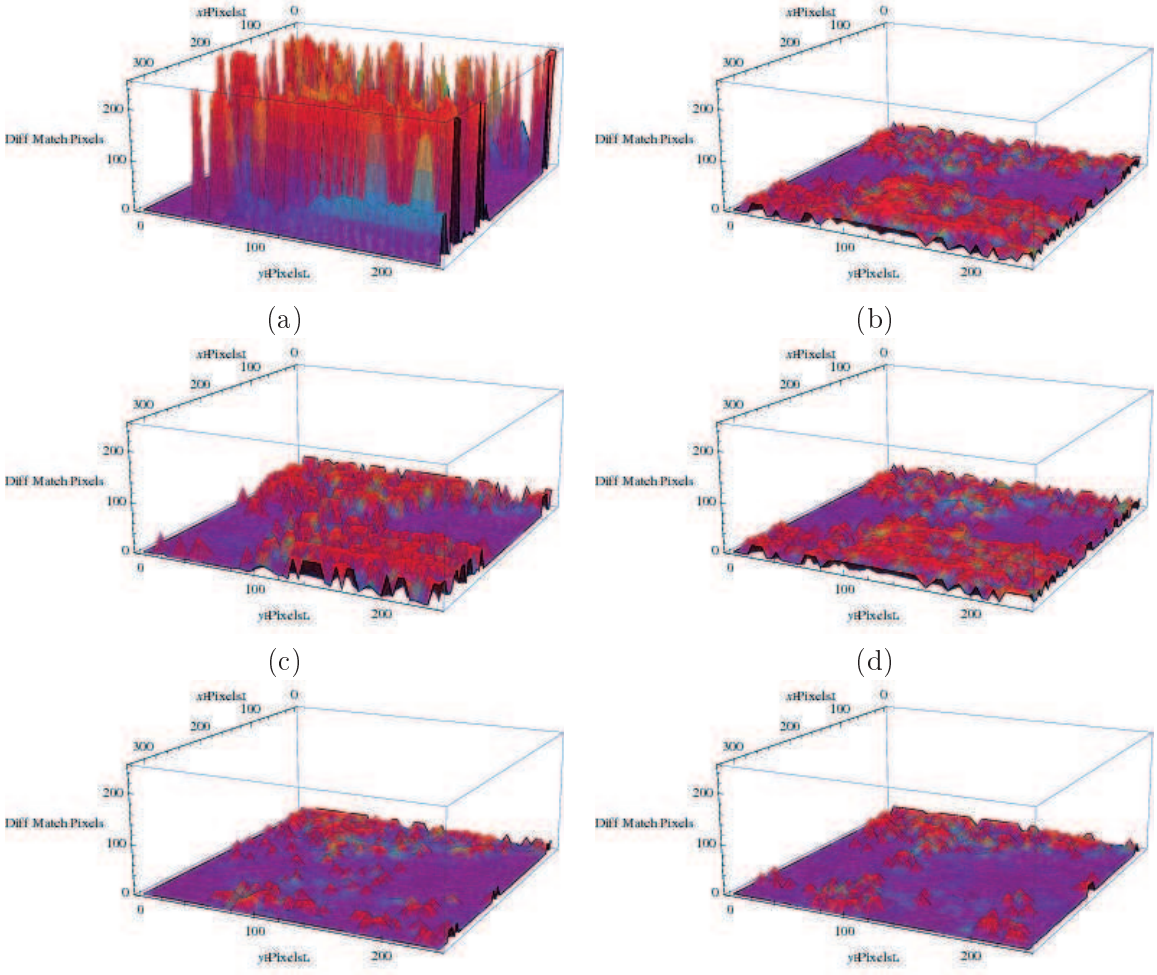


FIGURE 5.16 – Erreurs de recalage en variant le nombre de catégories de primitives. (a) et (b) erreur lorsque seule la première catégorie est employée. (c) et (d) : utilisation des catégories 1 et 2, l'erreur globale diminue. (e) et (f) : les catégories 1, 2 et 3 sont employées, l'erreur est très faible.

5.4 Décision cumulative 2D en cascade

Cette section décrit notre contribution au problème de la navigation d'un véhicule autonome doté d'un système de vision. La problématique est largement étudiée dans les domaines de la robotique et des systèmes d'aide à la conduite. En effet, la navigation sécurisée requiert au minimum une détection des obstacles potentiels (fixes ou mobiles) ainsi qu'une reconstitution de l'environnement même fruste. Afin d'accomplir ces tâches, il est acquis que le véhicule ou le robot devra se munir d'une multitude de capteurs (extéroceptifs tels que Radar et Lidar ou proprioceptifs tels que accéléromètres, gyromètres ou odomètres). Malheureusement, la plupart de ces capteurs fournissent des données entachées d'erreur et imprécises et peuvent être ponctuellement défaillants entrainant ainsi des données manquantes. Une idée naturelle est donc de faire coopérer ces capteurs, d'où l'essor considérable des techniques de fusion de données multi-capteurs [LBCT03], [CMR10]. Parmi tous les capteurs envisagées, la "caméra" a sans aucun doute une place particulière : malgré la complexité des processus de vision, la richesse des informations fournies ainsi que son coût en font un moyen privilégié, concentrant ainsi les efforts d'une grande partie de la communauté scientifique dans ces domaines [Dic02]. La profusion de publications récentes sur ces sujets en témoigne[8]. Le manque de recul immédiat, la multiplicité et la variété des approches proposées entraine la rareté voire l'inexistance de synthèses complètes récentes de techniques existantes. Sans être exhaustive, nous tenterons une première classification permettant de situer notre approche.

Approches 2D (x, y), basées "modèle" L'image 2D issue d'un capteur contient à elle seule un certain nombre d'informations sur l'environnement perçu. En particulier si l'objectif est d'éviter des obstacles et que ceux ci sont visuellement discriminants – dans le sens où l'on peut définir des signatures/attributs visuels les distinguant d'autres "objets" de l'environnement – alors il est possible de construire des modèles permettant de les détecter. Plusieurs approches ont été proposées, exploitant la symétrie [BBB+01], la texture [KTS98], ou la couleur [BD98]. Ces approches assez performantes pour des obstacles de type "véhicule" peinent à détecter ou "reconnaitre" un piéton dont la variabilité et les déformations rendent la tâche plus complexe. C'est tout naturellement que des techniques basées sur de la classification et de la reconnaissance ont été adoptées. Parmi les approches proposées, citons : les techniques de classifications binaires en cascade [GB00] reposant sur l'utilisation d'un classifieur, généralement linéaire ; l'utilisation des Machines à Vecteurs de Support [Vap99] ou le recours à des réseaux de neurones [MG06]. L'avantage de ces méthodes est leur capacité à travailler sur des espaces de grande dimension. Citons par ailleurs les techniques de boosting qui consistent à agglomérer plusieurs classifieurs faibles en un classifieur fort. Les résultats obtenus par le classifieur fort sont alors supérieurs à ceux obtenus par chaque classifieur faible. Les méthodes couramment utilisées sont dérivées de la méthode AdaBoost [HTFF05]. L'utilisation d'un tel classifieur repose sur la définition d'une base de représentation dans laquelle projeter les imagettes sur lesquelles vont s'effectuer les tâches d'entraînement et de détection. Une solution extrêmement populaire pour la détection de piétons est l'utilisation d'histogrammes de gradients orientés (Histograms

8. Des revues et conférences entières sont dédiées à ce sujet : ITSC, IV, IROS, IEEE Trans on Intelligent Transportation Systems, etc.

of Oriented Gradients, HOG) [DT05]. D'autres descripteurs ont été proposés, comme les Joint Ranking of Granules [HN10], la décomposition en ondelettes de HAAR [VJS05], ou encore une analyse en composantes principales [TP91]. Ces méthodes de détection reposent également sur une phase d'apprentissage hors ligne. Cet apprentissage se déroule en présentant au classifieur des populations de négatifs et de positifs. La représentativité des bases d'apprentissage va alors fortement conditionner l'aptitude du classifieur à discerner différents objets.

Approches 3D (x, y, z), basées "structure" Ces approches se basent sur l'estimation d'informations structurelles caractérisant les obstacles potentiels en exploitant une deuxième caméra qui assure ainsi un processus de vision stéréoscopique estimant la profondeur z des objets perçus. Dans certains travaux, les obstacles sont considérés comme étant des plans fronto-parallèles aisément détectables en particulier si le système de vision stéréoscopique est parfaitement calibré et rectifié. Dans [LAT02] un espace dans lequel les plans fronto-parallèles sont transformés en droites est défini. Celles-ci sont ensuite extraites à l'aide d'une transformée de Hough. Nous reviendront largement sur cette technique qui a inspiré les travaux de cette section. Au delà de la simple détection de plans fronto-parallèles, la construction de cartes, ou plus précisément de grilles d'occupation connaît un certain succès [VBA08, NBT09]. L'un des principaux intérêts de cette approche est que la collaboration entre plusieurs capteurs est alors immédiate et de type "tableau noir" (i.e. accumulation naturelle pour partage). En effet, une même carte d'occupation peut être peuplée en utilisant indifféremment des points issus d'un LIDAR, d'un RADAR ou de la stéréovision. D'une manière plus générale, la collaboration entre LIDAR et stéréovision est une piste fréquemment envisagée. Ainsi, le LIDAR fournira des hypothèses de détection que la vision viendra ensuite confirmer [RFBC10]. Ce problème de localisation des obstacles peut également être abordé par son dual : l'identification de l'espace libre devant le véhicule. La problématique n'est plus alors de chercher à éviter les menaces potentielles, mais de chercher à définir l'espace dans lequel il est possible pour l'égo-véhicule de manoeuvrer [SPA07].

Approches 3D (x, y, t), basées "mouvement" Avant de situer notre travail parmi les approches existantes et afin de mettre en évidence les différents points de vue adoptés, formalisons la problématique de la navigation d'un véhicule doté d'une seule caméra embarquée, le formalisme peut être facilement généralisé à plusieurs caméras.

Considérons le système de coordonnées $OXYZ$ calé au centre optique d'une caméra. L'axe OZ coïncide avec l'axe optique. Si l'on considère un mouvement rigide du capteur, caractérisé par sa vitesse translationnelle instantanée $\mathbf{T} = (T_X, T_Y, T_Z)$ et sa vitesse rotationnelle instantanée $\Omega = (\Omega_X, \Omega_Y, \Omega_Z)$, chaque point $\mathbf{P} = (X, Y, Z)$ appartenant à la scène statique est doté d'un mouvement relatif $\mathbf{V} = -\mathbf{T} - \Omega \times \mathbf{P}$. Si l'on considère que la projection[9] du point $\mathbf{P} = (X, Y, Z)$ dans le plan image est $p = (x, y, z)$, que la distance focale est f alors la vitesse 2D (u, v) en chaque point de l'image est :

9. On considère ici un modèle projectif simple (modèle sténopé).

$$\begin{cases} u = \frac{xy}{f}\Omega_X - \left(\frac{x^2}{f} + 1\right)\Omega_Y + y\Omega_Z - \frac{fT_X + xT_Z}{Z} \\ v = -\frac{xy}{f}\Omega_Y - \left(\frac{y^2}{f} + 1\right)\Omega_X - x\Omega_Z - \frac{fT_Y + yT_Z}{Z} \end{cases} \tag{5.4}$$

L'examen de ces équations permet plusieurs constatations :

- Le mouvement 2D dépend de la profondeur.
- Seule la composante translationnelle du mouvement dépend de la profondeur.
- Toute discontinuité de mouvement 2D ne peut être due qu'à une variation de profondeur.
- Le mouvement ne peut être déterminé qu'à un facteur d'échelle près. Un objet situé à une distance Z, se translatant de T produira le même mouvement 2D qu'un objet situé à une distance $2Z$, se déplaçant en translation de $2T$.

Devant ces constatations et la non linéarité des équations, certaines approches partent d'un modèle simplifié de projection [Alo90]. Sous les hypothèses de longueur focale importante et d'objets proches de l'axe optique, le modèle orthographique (ou perspective faible) considère que la projection 3D->2D se résume en une projection orthogonale suivie d'un changement d'échelle. Dans le cas où les objets sont éloignés de la caméra ou de petite dimension par rapport à la distance au centre de projection, l'emploi de la projection paraperspective permet aussi de simplifier les équations [PK92]. Sans passer par des hypothèses sur la nature de la scène, il est possible aussi de faire usage de la projection sphérique bien adaptée à la représentation des champs de vecteurs.

A côté de cette tentative de simplifier les équations à travers le modèle de projection, certains travaux préfèrent simplifier le modèle du mouvement lui même, en considérant par exemple un véhicule mobile en translation longitudinale majoritaire pure ou en introduisant un ou deux angles de rotation (souvent le lacet pour la prise en compte des virages) [BDW06], [SFS09]. D'autres approches tentent non pas de limiter le nombre de degrés de liberté mais de séparer l'estimation des translations (dépendant de la profondeur) des rotations en exploitant la parallaxe que crée ce mouvement (*motion parallax*, *affine motion parallax*, *plane+parallax*). Ces méthodes exploitent le fait qu'aux discontinuïtés de profondeur, il est possible de distinguer les effets de la rotation de ceux de la translation de la caméra. En particulier pour les approches type "Plane+parallax", le mouvement 2D d'une région de l'image où les variations en profondeur ne sont pas significatives permet de supprimer les effets de la rotation de la caméra. A partir de ce mouvement de parallaxe résiduel obtenu, la translation peut être facilement calculée [IRP97], [HKM08]. Dans le même esprit de simplification du modèle de mouvement, citons les approches se concentrant sur l'estimation du Foyer d'Expansion dont on sait que les coordonnées regroupent les informations liées au mouvement translationnel [XD92], [SRR04], [FWH07].

Ces dernières années se sont multipliés les travaux sur l'estimation du mouvement 3D (*egomotion*) d'une caméra embarquée sur véhicule mobile et sur la reconstitution de la profondeur de la scène observée (*Structure From Motion*). Il s'en est suivi de nombreuses

classifications des méthodes existantes selon divers critères. La classification couramment adoptée distingue 3 catégories principales : les approches discrètes, continues et directes.

- Les approches **discrètes** se basent sur la mise en correspondance de primitives image et sur une expression matricielle, incluant toutes les inconnues (paramètres de mouvement et paramètres intrinsèques de la caméra), reliant les points de correspondance. Le problème se ramène alors à un problème d'algèbre linéaire et les nombreuses approches existantes diffèrent dans le choix des méthodes de résolution adoptées, plus ou moins sensibles aux perturbations des données [LF97], [Har95], [LHP80].
- Les approches **continues** exploitent le flot optique calculé. La relation entre le flot optique et le mouvement préalablement paramétré permet - par des techniques d'optimisation - d'estimer les paramètres du mouvement ainsi que la profondeur en chaque point. Les résultats obtenus sont alors dépendants de la qualité du flot optique calculé [MJF94], [Hil92], [NH89].
- Dans les approches **directes**, le mouvement est déterminé "directement" à partir de la contrainte d'invariance de la luminosité d'un point au cours de son déplacement sans avoir à calculer explicitement le flot optique. Les paramètres du mouvement sont alors déduits par des approches d'optimisation classiques. Les approches **continues** et **directes** sont toutes deux globales et récoltent des informations sur toute l'image. Leur avantage par rapport aux approches discrètes provient alors du nombre important de données traitées, contribuant ainsi à réduire les erreurs [SMS00], [IRP97].

Approches 4D (x, y, z, t), basées coopération "structure/mouvement" L'idée de faire collaborer estimation du mouvement et estimation structurelle n'est pas neuve. Les travaux faisant intervenir activement ces deux approches se succèdent depuis le début des années 2000, c'est-à-dire depuis que la puissance de calcul disponible permet de mener ces deux processus de front. Parmi les travaux les plus significatifs, citons ceux décrits dans [Hei02] centrés sur l'exhibition d'un invariant de l'image, en l'occurence le rapport de la norme du flot optique sur la distance au capteur. Par ailleurs, le principe de 6D-Vision, avancé dans [FRBG05] constitue une approche intéressante. Elle repose sur le suivi de points d'intérêts en utilisant des filtres de Kalman, accordés sur les mouvements susceptibles d'animer les objets de la scène. Comme nous l'avons vu plus haut, le formalisme des grilles d'occupations permet une intégration aisée de différents capteurs. Il est donc naturel de le retrouver exploité ici afin de faire coopérer les différentes modalités de la vision artificielle. Ce formalisme peut être exploité afin de construire une représentation de la scène observée [DC00], ou simplement enrichi de l'information temporelle [BPU+08, LCCG07]. Les approches recevant le plus grand intérêt de la communauté sont cependant celles centrées sur l'évaluation du *scene-flow*, soit l'extension du flot optique à un espace tridimensionnel. Pour cela, il est possible de suivre des points d'intérêt [LZGR11] ou d'intégrer la stéréo dans une méthode de calcul du flot optique type Horn & Schunk [PKF07, WRV+08]. A partir de ce champ de correspondances, des techniques de segmentations classiques peuvent être utilisées pour obtenir une représentation de la scène en fonction du mouvement apparent des objets.

Notre contribution Nous avons proposé deux approches toujours dans le cadre de la vision "fruste" qui sert de fil rouge à nos travaux :

- L'une développée dans le cadre de la thèse d'Adrien Bak est une approche (3D+t) visant à concevoir un système d'estimation de l'égo-mouvement et de détection d'obstacles fondé sur la coopération entre le mouvement et la stéréovision. Nous montrons que le problème de l'estimation de l'ego-mouvement peut être posé de façon linéaire sans perte de précision et sans simplification du modèle de mouvement. Cette approche ne sera pas développée ici ; on se référera à nos publications : [BBA11a] et [BBA10].
- Dans l'autre, nous partons de l'hypothèse qu'une navigation sécurisée du véhicule requiert au moins un étiquetage fruste de la scène. Les objets sont étiquetés en fonction de leur nature en relation avec leur structure (surface horizontale = route ; surface verticale = bâtiment ; surface frontale = obstacle) et en fonction de leur mouvement (mouvement conforme à l'ego-mouvement du véhicule = objet statique ; mouvement non conforme à l'ego-mouvement du véhicule = objets ayant un mouvement indépendant). Nous considérons par ailleurs, que toutes les possibilités de la vision monoculaire, pourtant économe en moyens, n'ont pas encore été exploitées, ce qui classe notre approche parmi les approches 3D (x, y, t) permettant de fournir une estimation fruste de la structure. Partant de cet objectif, deux études dans la littérature ont retenu notre attention.
  - Dans la première, [FA95], les vecteurs vitesse d'amplitude et d'orientation données sont contraints d'appartenir à des courbes dans l'image dont les paramètres dépendent des paramètres du mouvement 3D du capteur. En particulier, si les vecteurs considérés sont issus d'un flot optique ou d'un champ de disparité alors on peut montrer que ces vecteurs sont contraints d'appartenir à des sections de coniques pouvant être définies. En étudiant les propriétés de ces courbes, une estimation de l'ego-mouvement peut être réalisée.
  - Dans la seconde étude [LAT02], les auteurs proposent, en stéréovision, une technique très efficace basée sur le concept de *v-disparité* qui consiste à exploiter la relation entre disparité et lignes image dans le cas particulier où les images stéréoscopiques sont rectifiées. Un nouvel espace de projection cumulatif – l'espace appelé *v-disparité* – composé des histogrammes de disparité sur toutes les lignes image, permet de mettre en évidence la relation de proportionnalité entre disparité et lignes image dans le cas particulier d'un plan horizontal.

  Ces deux études peuvent être mises en parallèle d'une manière intéressante : toutes deux exploitent des courbes d'iso-valeurs –vitesse pour l'une, disparité pour l'autre. Notre point de vue est que ce procédé, basé d'une part sur la définition d'iso-courbes et s'appuyant d'autre part sur des statistiques le long de ces courbes, peut être étendu. Nous avons donc naturellement pensé à sa généralisation au cas de la vision monoculaire.

Cette section décrit en détail l'approche proposée. Des surfaces paramétrées peuvent être détectées sans aucune calibration *a priori* de la caméra, ni aucune connaissance de l'ego-mouvement du véhicule. Dans un premier temps, dans le but de contourner l'estimation de la profondeur des objets (non connue), nous partons de l'hypothèse que la scène 3D peut être approximée par un ensemble de plans. Ces plans seront alors détectés et étiquetés en fonction de leur mouvement en exploitant les courbes d'iso-vitesse 2D, les vitesses pouvant être estimées par un flot optique quelconque. Pour tenir compte des imprécisions liées

à l'estimation du flot optique et parce que nous sommes convaincue de la robustesse des techniques cumulatives dans ce contexte applicatif, nous définissons un espace cumulatif appelé *c-vélocité* par analogie à la *v-disparité*.

5.4.1 Définition des entités à cumuler

Dans l'approche que nous proposons, l'estimation des vecteurs vitesse par une méthode de flot optique dense engendre une population de votants de taille conséquente rendant le processus de décision cumulatif représentatif et pertinent. Pour un modèle de mouvement et une structure de nature donnés (un plan d'orientation donnée et situé à une distance donnée) les "primitives" sont associées à une vitesse. Celle-ci se trouve sur une courbe d'iso-vitesse renforçant l'hypothèse du mouvement et de la structure conjecturés. L'espace de cumul est bi-dimensionnel (paramètre de courbe d'iso-vitesse / norme de vitesse) puis monodimensionnel car les vitesses le long de ces courbes sont non seulement constantes pour des structures définies mais elles sont aussi liées par une relation linéaire. Un deuxième espace cumulatif 1D est alors défini permettant d'extraire du premier espace les relations de proportionnalité (droite) exhibées.

Dans [LAT02], les auteurs prouvent que pour un plan horizontal, le long d'une ligne image, issue d'un couple d'images stéréoscopiques rectifiées, la disparité est constante et varie linéairement en fonction de la profondeur et donc des lignes image. Le plan de la route est alors détecté dans l'espace *v-disparité*, construit en accumulant les disparités le long des lignes image. La route se projette dans cet espace en une droite. Le procédé a été généralisé aussi aux plans verticaux en considérant les colonnes image et définissant par analogie, l'espace *u-disparité*.

Nous montrons dans ce qui suit comment le procédé peut être généralisé à la vitesse (déplacement en analogie à la disparité), cumulée le long de courbes d'iso-vitesse (en analogie aux lignes et colonnes image).

Cas d'un point 3D en mouvement : Considérons dans un premier temps un mouvement translationnel le long de l'axe Z (le véhicule avance). Nous verrons par la suite comment le généraliser à d'autres types de mouvements. En se référant à l'équation du mouvement d'un point (voir Equation (5.4)) et en posant $\Omega_X = \Omega_Y = \Omega_Z = T_X = T_Y = 0$, la vitesse 2D (u, v) devient :

$$\begin{cases} u = \frac{T_Z}{Z} x \\ v = \frac{T_Z}{Z} y \end{cases} \tag{5.5}$$

Les équations (5.5) décrivent le mouvement 2D d'un point (projection dans l'image du mouvement 3D) qui ne doit pas être confondu avec le flot optique. Nous partirons de l'hypothèse communément admise que le flot optique est une approximation correcte du mouvement 2D.

Considérant l'équation (5.5), la relation entre la vitesse $\|\mathbf{w}\|$ (analogie avec la disparité) et la fonction iso-vitesse c (analogie avec les index ligne v) devient :

$$\|\mathbf{w}\| = \sqrt{u^2 + v^2} = \left|\frac{T_Z}{Z}\right| \sqrt{x^2 + y^2} = K \cdot C(x, y) \tag{5.6}$$

$$\frac{\|\mathbf{w}\|}{K} = C(x, y) = c \tag{5.7}$$

Prenons pour illustrer le concept le cas où la profondeur Z est constante en tout point (i.e. le véhicule s'approche d'un mur par exemple), la translation T_Z étant celle de la caméra, identique pour tous les points statiques. De ce fait, K, défini par $\left|\frac{T_Z}{Z}\right|$ dans l'équation (5.6) est constant, et les courbes d'iso-vitesse $C(x, y)$ sont des cercles. De plus, le rayon de ces cercles (paramètre des courbes d'iso-vitesse) c varie linéairement avec la norme de la vitesse $\|\mathbf{w}\|$ (voir equation (5.7)). A côté de ce cas particulier (profondeur constante) décrit ici pour illustrer la démarche, dans le cas général, Z peut être simplement éliminé en considérant des surfaces planaires permettant ainsi d'exploiter la relation entre Z et (X, Y) pour éliminer Z.

Cas d'un plan 3D en mouvement : Supposons à présent que la caméra observe une surface plane d'équation $\mathbf{n}^T\mathbf{P} = d$, avec $\mathbf{n} = (n_X, n_Y, n_Z)$ le vecteur unitaire normal à la surface, d la distance "plan/origine" et P le point de coordonnées (X, Y, Z). A partir de l'équation (5.4) et de $Z = \frac{1}{n_Z}(d - n_X X - n_Y Y)$, la vitesse 2D s'écrit alors [LHP80, VP89] :

$$\begin{cases} u = \frac{1}{fd}\left(a_1 x^2 + a_2 xy + a_3 fx + a_4 fy + a_5 f^2\right) \\ v = \frac{1}{fd}\left(a_1 xy + a_2 y^2 + a_6 fy + a_7 fx + a_8 f^2\right) \end{cases} \tag{5.8}$$

$$\begin{aligned} a_1 &= -d\Omega_Y + T_Z n_X \\ a_2 &= d\Omega_X + T_Z n_Y \\ a_3 &= T_Z n_Z - T_X n_X \\ a_4 &= d\Omega_Z - T_X n_Y \\ a_5 &= -d\Omega_Y - T_X n_Z \\ a_6 &= T_Z n_Z - T_Y n_Y \\ a_7 &= -d\Omega_Z - T_Y n_X \\ a_8 &= d\Omega_X - T_Y n_Z \end{aligned}$$

Etudions précisément 4 cas particuliers de plans en mouvement adaptés à l'application visée :

a) Horizontal (route)

b) Latéral (bâtiment)

c) Frontal1 (obstacle fuyant ou approchant)

d) Frontal2 (obstacle traversant)

La TABLE 5.1 liste, pour chaque cas, le vecteur normal unitaire associé au plan $\mathbf{n}$, le vecteur translation 3D $\mathbf{T}$ et la distance plan-origine d. Le mouvement caméra est supposé translationnel $\mathbf{T} = (0, 0, T_Z)$. En conséquence, dans l'équation (5.8), $T_X, T_Y, \Omega_X, \Omega_Y$ et Ω_Z sont annulés (à l'exception de l'obstacle fuyant/approchant possédant sont propre modèle de mouvement $(0, 0, T_Z^o)$ ou $(T_X^o, 0, 0)$ qui s'ajoute à celui de la caméra $\mathbf{T}$ pour donner : $\mathbf{T}'$).

TABLE 5.1 – Paramètres de 4 types de plans

	$\mathbf{n}$	mouvement 3D	Dist. plan-origine
a)	$(0, 1, 0)$	$\mathbf{T} = (0, 0, T_Z)$	d_r
b)	$(1, 0, 0)$	$\mathbf{T} = (0, 0, T_Z)$	d_b
c)	$(0, 0, 1)$	$\mathbf{T}' = (0, 0, T_Z^o + T_Z)$	d_o
d)	$(0, 0, 1)$	$\mathbf{T}' = (T_X^o, 0, T_Z)$	d_o

Les vecteurs vitesse correspondants sont ainsi obtenus en injectant respectivement $\mathbf{T}$ (ou $\mathbf{T}'$) et $\mathbf{n}$ dans a_i for $i = 1, ..., 8$. L'équation (5.8) donne alors u et v comme listé dans la TABLE 5.2 pour chaque cas.

Soient $\|\mathbf{w}_o\|, \|\mathbf{w}_r\|$, et $\|\mathbf{w}_b\|$, amplitudes des vitesses associées respectivement à un obstacle, une route et un bâtiment. Regroupons les paramètres du mouvement 3D et la focale inconnus dans le paramètre K que nous ne chercherons pas à estimer ici ; son seul intérêt à ce stade est d'être constant.

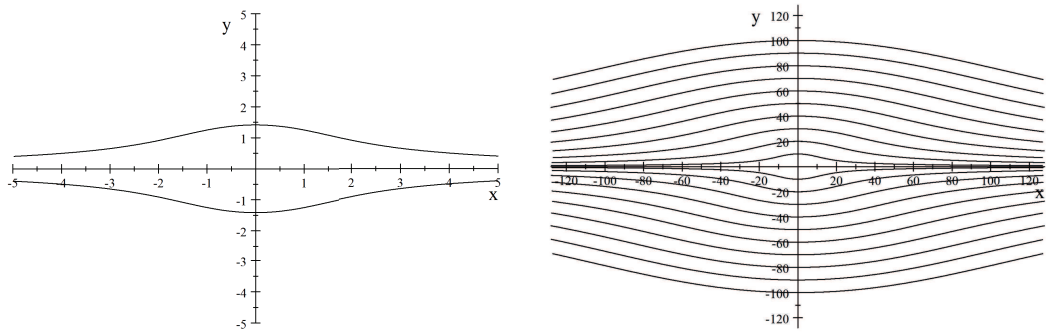
TABLE 5.2 – Vecteurs vitesse associés à 4 types de plan

a)	$u = \frac{T_Z}{fd_r} xy$ $v = \frac{T_Z}{fd_r} y^2$	$\|\mathbf{w}_r\| = K\sqrt{y^4 + x^2y^2}$
b)	$u = \frac{T_Z}{fd_b} x^2$ $v = \frac{T_Z}{fd_b} xy$	$\|\mathbf{w}_b\| = K\sqrt{x^4 + x^2y^2}$
c)	$u = \frac{T_Z + T_Z^o}{d_o} x$ $v = \frac{T_Z + T_Z^o}{fd_o} y$	$\|\mathbf{w}_o\| = K\sqrt{x^2 + y^2}$
d)	$u = \frac{T_Z}{d_o} x - \frac{T_X^o f}{d_o}$ $v = \frac{T_Z}{d_o} y$	$\|\mathbf{w}_o\| = \begin{cases} K \text{ if } T_X^o \ggg T_Z \\ K\sqrt{x^2 + y^2} \text{else} \end{cases}$

Mise en évidence de la relation de proportionnalité $c/\|\mathbf{w}\|$: Chaque type de $\|\mathbf{w}\|$ conduit à une expression de c et de ce fait à des courbes d'iso-vitesse différentes résumées dans la TABLE 5.2, de a) à d). Par souci de clarté, nous emploierons le terme $c - courbe$ pour "courbe d'iso-module de vitesse" et le terme $c - valeur$ pour la valeur $c(x, y)$ le long d'une $c - courbe$. En particulier, dans le cas b) correspondant à un plan bâtiment,

$$c = \frac{\|\mathbf{w}\|}{K} = \sqrt{x^4 + x^2y^2} \tag{5.9}$$

La FIGURE (5.17 (a) illustre les $c - courbes$ pour une valeur c_0 donnée de c, pour un "plan horizontal" : cas a) dans la TABLE 5.2). Chaque courbe est l'ensemble des pixels dont l'amplitude de vitesse est constante, $\mathbf{w} = Kc_0$, si et seulement si les points appartiennent bien à l'image d'un plan horizontal. Par conséquent, la relation précédente (5.9) prouve que c, constante le long d'une courbe d'iso-vitesse par définition, est proportionnelle à $\|\mathbf{w}\|$. En réalité, un plan horizontal intersecte dans l'image la famille de ces courbes obtenues en variant c : voir FIGURE (5.17 (b) où sont affichées les courbes en incrémentant c de 10.



(a) (b)

FIGURE 5.17 – (a) Un couple de courbes pour une valeur de c donnée dans le cas "plan horizontal". L'accumulation est faite le long de ces courbes qui ne s'intersectent pas en théorie. (b) Un ensemble de $c - courbes$ pour un pas de variation égal à 10.

Rectifications des $c-courbes$: A chaque point $\mathbf{p} = (x, y)$ dans l'image, est associée une valeur c en fonction du modèle de plan choisi. Cette valeur peut être calculée hors ligne une seule fois à l'initialisation puisqu'elle ne dépend que de (x, y). Par ailleurs, il est possible pour faciliter l'implémentation de l'approche et par analogie à la rectification d'images stéréo, de calculer la transformation permettant de redresser les courbes d'iso-vitesse afin de les rendre parallèles aux lignes ou au colonnes image en fonction du modèle considéré. Les images résultantes ne sont autres que $I(c, y)$ pour les modèles route et obstacle et $I(x, c)$ pour le modèle bâtiment. Le détail des calculs est donné dans [BZ11a].

Extraction du Foyer d'expansion : L'origine du repère image – servant notamment au calcul des $c - valeurs$ – est l'intersection du plan de projection et de la direction de translation. Ce point particulier appelé Foyer d'Expansion (FoE) doit être déterminé le plus précisément possible : son estimation est donc une étape clé du processus visuel décrit ici. Dans le cas d'un mouvement translationnel, chaque vecteur vitesse se dirige vers le FoE : voir Equation (5.5).

$$\frac{u}{v} = \frac{x}{y} \tag{5.10}$$

Supposons que (x_0, y_0) sont les coordonnées du FoE dans l'image. Sachant que l'origine du repère image est placé en haut à gauche, alors les équations de l'ego-mouvement (u, v) translationnel deviennent :

$$\begin{cases} u = \frac{T_Z}{Z}(y_0 - y) \\ v = \frac{T_Z}{Z}(x - x_0) \end{cases} \tag{5.11}$$

et

$$\theta = \tan^{-1}\left(\frac{v}{u}\right) = \tan^{-1}\left(\frac{x - x_0}{y_0 - y}\right) \tag{5.12}$$

Cette relation montre que le FoE peut être extrait simplement en déterminant l'intersection des droites supportant les vecteurs vitesse. Plusieurs approches ont été proposées dans la littérature [SRR04], [NH89]. Nous avons choisi de maintenir une cohérence avec le point de vue "cumulatif" pour des raisons de vision fruste incluant notamment la réutilisation de modules [BZ06]. Dans l'approche implémentée, les vecteurs vitesse votent pour tous les points appartenant à la droite les supportant. Le FoE résultant est alors le point qui recueille le maximum de votes, c'est-à-dire l'intersection de toutes les droites supportant les vecteurs vitesse : voir FIGURE 5.20, images étiquetées (b). Dans ces images : en rouge sont représentés les points qui recueillent le maximum de votes ; l'intersection des deux droites en blanc donne la position du FoE estimé ; le taux de votant pour chaque point est représenté par une couleur allant du noir (aucun vote) au rouge (nombre de votants maximum) ; les taux intermédiaires sont respectivement colorés en bleu, cyan, vert et jaune.

La position du FoE peut confirmer ou non l'hypothèse de mouvement translationnel du capteur. Par exemple, si le FoE n'est pas au centre de l'image, les images peuvent être rectifiées pour compenser les effets éventuels des roulis ou tangages de la caméra [BZ11c]. Notons par ailleurs que l'équation (5.10) peut aussi servir à ajouter des contraintes additionnelles au moment du vote dans l'espace *c-vélocité*.

5.4.2 Description de l'espace cumulatif *c-vélocité*

Nous avons montré que la vitesse 2D est constante le long des courbes d'iso-vitesse (les $c-courbes$), caractérisées chacune par un paramètre constant c, correspondant au modèle de plan 3D adéquat.

Si l'on considère toutes les perturbations pouvant affecter les amplitudes des vitesses estimées, il est peu probable d'obtenir des vitesses constantes le long de ces courbes. Nous proposons donc de considérer le mode à travers une analyse de l'histogramme des $\|\mathbf{w}\|$ le long des $c-courbes$. En collectant ainsi tous les modes le long de toutes les $c-courbes$, la relation de proportionnalité entre $\|\mathbf{w}\|$ et c est mise en évidence. Par conséquent, l'espace *c-vélocité*, **bidimensionnel** en $(c, \|\mathbf{w}\|)$, est cumulatif : il est construit en affectant à chaque pixel (x, y) la valeur c ($c - valeur$) correspondant au modèle de plan choisi –a), b), c) ou

d) dans la TABLE (5.2)– et en incrémentant la valeur au coordonnées $(c, \|\mathbf{w}\|)$, où $\mathbf{w}$ est la vitesse estimée en (x, y). Nous avons choisi dans nos expérimentations d'estimer $\mathbf{w}$ par une approche classique de calcul de flot optique, en l'occurrence celle proposée par Lucas & Kanade [LK81].

Considérations numériques : Une étude de la fonction $c(x, y)$ pour chaque modèle de plan –en particulier pour le modèle "route" et le modèle "bâtiment"– nous amène aux conclusions suivantes : d'abord, chacune de ces courbes intersecte l'axe des x (pour le modèle route) ou l'axe y (pour le modèle bâtiment) dans le plan image aux coordonnées : $x = \pm\sqrt{c}$ ou $y = \pm\sqrt{c}$, respectivement. Par ailleurs, pour une taille d'images standard, l'intervalle de variation de c est très grand, en l'occurrence égal à 128000 (pour le modèle route) et à 96000 (pour le modèle bâtiment) pour une image de taille 320×240. Par conséquent, pour des raisons d'implémentation, de complexité autant que d'homogénéité, nous choisissons pour ces deux modèles de plans de considérer plutôt la relation entre $\|\mathbf{w}\|$ et $\sqrt{c}$[10]. Un plan est alors représenté dans l'espace *c-vélocité* par une parabole au lieu d'une droite.

5.4.3 Décision

Nous avons montré comment un plan 3D était représenté dans l'espace *c-vélocité* par une parabole. La détection de ces paraboles dans l'espace *c-vélocité* pour leur retro-projection dans l'image, nous permet de déterminer effectivement les plans 3D associés. La détection des paraboles dans un espace 2D peut se faire de différentes manières ; nous avons encore une fois privilégié le choix d'une méthode cumulative. C'est donc naturellement que nous sommes tournée vers la transformée de Hough. Ici, elle sera 1D car les paraboles passent par l'origine du repère (les $c - courbes$ sont définies à partir de l'origine i.e. le FoE). L'espace de Hough 1D est construit en cumulant le paramètre p de chaque parabole, i.e. la distance p entre chaque parabole et son foyer ou sa directrice [DH72]. L'ensemble du processus obéit aussi à l'expression fonctionnelle suivante, témoin de la réutilisation de procédé et de l'économie des moyens :

$$(x, y, \|\mathbf{w}\|) \rightarrow (c, \|\mathbf{w}\|, P(c, \|\mathbf{w}\|))$$

$$\rightarrow [p(c, \|\mathbf{w}\|), \textstyle\sum_{[(c,\|\mathbf{w}\|),p(c,\|\mathbf{w}\|)]} P(c, \|\mathbf{w}\|)] \; [i.e.(p(c, \|\mathbf{w}\|), P(p))]$$

où P est la probabilité et p le paramètre de la parabole. L'histogramme 1D obtenu est alors segmenté par une méthode de clustering quelconque, ici un K-means classique [Mac03].

$$\|\mathbf{w}\| = K\left(\sqrt{c}\right)^2 \Rightarrow p = \frac{1}{4K} = \frac{\left(\sqrt{c}\right)^2}{4\|\mathbf{w}\|} \tag{5.13}$$

10. D'autres fonctions servant à réduire la dynamique peuvent être employées.

En théorie, chaque plan 3D correspond à une parabole (une valeur donnée du paramètre p) dans l'espace *c-vélocité*. Il est évident que plusieurs perturbations concourent à transformer une parabole (un Dirac dan l'espace de Hough 1D) en une patatoïde parabolique (une Gaussienne dans l'espace de Hough 1D). Nous les avons étudiées en détail dans [BZ11c]. C'est la raison pour laquelle l'espace de Hough 1D doit être segmenté en clusters. On peut objecter que l'utilisation des K-means impose de fixer le nombre de classes *a priori*. Nous pouvions aisement nous orienter vers des méthodes de clustering non supervisées. Cependant, dans le cadre des applications traitées et à nouveau en conformité avec la vision fruste, le nombre de clusters peut être fixé *a priori*. On peut imaginer que le nombre de plans est limité par la structure et la nature de la scène. En particulier, dans les scènes urbaines nous pouvons dénombrer 4 types de plans : 1) horizontal (une route), 2) verticaux (2 bâtiments de part et d'autre du véhicules et éventuellement 2 plans correspondant aux plans des voitures garées de part et d'autre de la route, ce qui donne 4 classes dans cette catégorie), 3) des obstacles fronto traversants et 4) des obstacles frontaux fuyants/approchants . Il faut être conscient des conséquences d'un tel *a priori* :

- Supposons que le nombre de plans est supérieur au nombre de classes fixées dans le K-means clustering. Certains classes d'orientations proches vont donc être fusionnées. La question est alors de savoir si cela entraîne des conséquences sur la précision pour la navigation d'un véhicule ? (ex : distance entre le véhicule et les plans latéraux). Cette question est abordée plus précisément dans [BZ11c].
- Peut-on, indépendamment du nombre de plans, prévoir une correction *a posteriori* ? Nous pouvons certainement envisager de reboucler sur le résultat du K-means en affinant le clustering en fonction de la vraisemblance du résultat. Par exemple dans l'image de la FIGURE 5.20, prévoir une classe supplémentaire permet de séparer le plan "lampadaire" du plan "voitures garées à droite".
- Ce problème est un dilemme classique entre un clustering non supervisé plus complexe et un plus élémentaire avec connaissance *a priori*. Dans cette étude, nous avons privilégié le second choix, le but étant davantage de valider d'abord le principe de la méthode.

Remarque : Les plans fronto-traversants sont représentés par des droites verticales et ce quel que soit le plan *c-vélocité* considéré. (voir FIGURE 5.19), parce que leur vitesse est quasi-constante.

Le schéma de la FIGURE 5.18 résume les différentes étapes de l'approche proposée.

5.4.4 Résultats de la détection des plans

5.4.4.1 Images synthetiques

Dans le cas d'école de la FIGURE 5.19 (a), un champ de vecteur vitesse synthétique a été généré. Il correspond à une scène 3D composée de 3 plans : un plan vertical (à gauche), un plan horizontal (en bas) et un plan frontal avec ses propres paramètres de mouvement (obstacle traversant).

Les résultats sur flots synthétiques confirment bien la transformation des plans considérés en paraboles dans les espaces *c-vélocité* dédiés. Dans la FIGURE 5.19 (b), la parabole

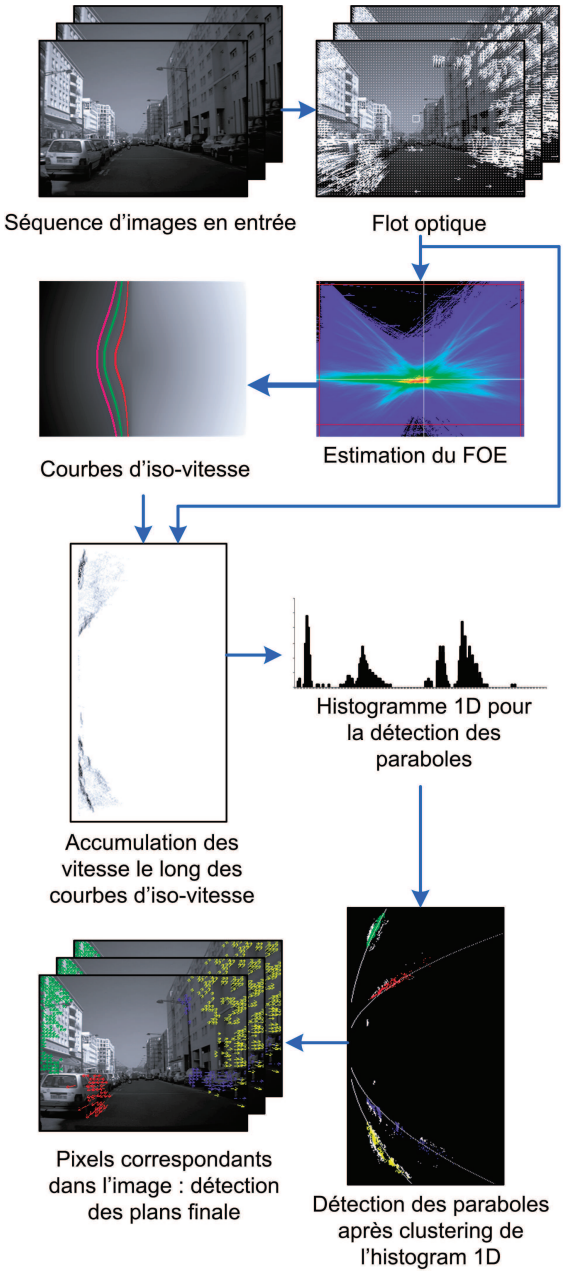


FIGURE 5.18 – Schéma récapitulatif de l'approche proposée.

partielle indique le plan attendu, sa taille étant proportionnelle à la taille du plan dans l'image. Notons l'effet de *moiré* dû aux perturbations inter-modèles constituées des votes minoritaires de la route et de l'obstacle dans l'espace bâtiment. Dans la FIGURE 5.19 (c), la parabole est complète car la route s'étale de part et d'autre du FoE. Notons l'effet de *moiré* cette fois moins prononcé en raison du nombre réduit de votants provenant d'autres modèles (la taille du bâtiment est plus réduite que celle de la route). L'obstacle traversant apparaît dans les deux espaces de vote comme un segment vertical (cf. *Remarque* ci-dessus).

5.4.4.2 Images réelles

Données et paramètres en entrée : Toutes les séquences d'images ci-dessous sont issues de la base de données constituée dans le cadre du projet ANR LOVe (Logiciel d'Observation des VulnérablEs). Ces séquences sont très variées et ont été fournies par les constructeurs automobiles partenaires du projet. L'approche proposée requiert en entrée un flot optique. Nous avons choisi la méthode classique de Lukas & Kanade [LK81] avec 9×9 comme taille de fenêtre d'analyse. Nous avons de manière délibérée choisi une approche d'estimation du flot optique la plus basique et classique qui soit, car notre objectif est, rappelons le, de confirmer le maintien de la robustesse de la méthode de détection de plan quelle que soit la qualité du flot en entrée. Les résultats présentés ci-dessous se focalisent sur la détection de plans latéraux qui cadrent bien avec les scènes urbaines considérées et l'application de conduite automatique.

Deux paramètres ont été introduits. Le premier est le taux minimum de votants dans une cellule de l'espace *c-vélocité*. Ce taux minimum est calculé en fonction du nombre maximum de points le long de courbes c. En raison de la discrétisation des courbes, celles-ci peuvent en effet avoir un nombre variable d'éléments. Le second paramètre est le nombre minimum de points votant pour un paramètre de parabole donné p dans l'espace de Hough 1D : trivialement le seuil est ici égal à 3 points. Par ailleurs, afin d'évaluer quantitativement notre approche, deux facteurs de confiance ont été définis. Le premier est lié à l'hypothèse de mouvement translationnel : il s'agit de la différence Δ_{foe} entre la position du FoE trouvé et le centre de l'image. Si Δ_{foe} est grand, le véhicule n'est pas en translation pure. Le second est l'écart type σ des classes trouvées après K-means. Un σ faible ajouté à un pic élevé

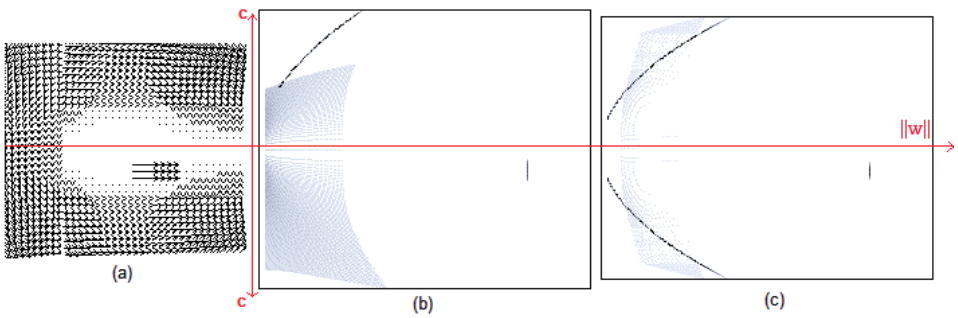


FIGURE 5.19 – a) Champs des vecteurs vitesse correspondant à une scène composée d'un bâtiment, d'une route et d'un obstacle traversant. b) et c) Les espaces *c-vélocité* associés (gauche : espace bâtiment ; droite : espace route).

dans l'espace de Hough 1D confirme la détection d'un plan. En effet, le nombre de points constituant le plan est alors le nombre de votants.

Resultats : Les exemples de la FIGURES 5.20 illustrent les images de flot optique (images étiquetées "a") échantillonnées par souci de lisibilité, la position du FoE (images "b") et l'espace *c-vélocité* "bâtiment" associé à différentes séquences d'images (images "c"). Les résultats de la détection des paraboles sont donnés dans les images étiquetées "d" avec les plans correspondants dans les images "e".

Interprétation : Dans toutes les séquences considérées, le cercle noir est le centre de l'image. Δ_{foe} est sa distance par rapport au FoE. Dans les images 1 et 2, on peux dénombrer six plans 3D : 2 plans "bâtiment", 2 plans correspondant aux voitures garées de part et d'autre de la route, un plan frontal traversant (une moto) et un plan "route". Dans cette séquence, la plupart des vecteurs vitesse estimés sont situés sur les plans latéraux qui sont plus larges et texturés. Dans les espaces *c-vélocité* "bâtiment" dans (d.1) et (d.2), comme attendu, 4 paraboles correspondant aux 4 plans latéraux sont nettement visibles. Les plans en (e.1) et (e.2) sont étiquetés en fonction du résultat des K-means. La même étiquette (et couleur) est utilisée pour afficher les points correspondants dans l'image. Les points écartés sont en blanc. L'image 3 montre un exemple où la caméra n'est pas en mouvement ; des piétons traversent la rue ; l'espace *c-vélocité* correspondant (c.3) fait apparaître une verticale (vitesse constante).

Dans l'exemple de la FIGURE 5.21, nous montrons comment l'approche peut être employée pour détecter des obstacles traversants à moindre coût. En sélectionnant les points non retenus (non étiquetés) dans l'espace *c-vélocité* "bâtiment" et en considérant seulement ceux localisés entre les paraboles (cadrés par deux murs des bâtiments droite et gauche), on cherchera les points alignés verticalement en même temps dans les trois espaces de c-vélocité bâtiment, route et obstacle fuyant/traversant. Les pixels dans l'image ayant voté pour ces points sélectionnés sont affichés en rose, les rectangles englobants associés sont donnés dans les images f de la même figure.

D'autres expérimentations ont été réalisées en faisant varier le nombre de plans dans l'image, lorsque la caméra est en rotation autour de l'axe Y (virage), lorsque un camion par exemple traverse la route ou lorsqu'un véhicule se déplace avec approximativement le même mouvement relatif. Tous les résultats sont disponibles dans [BZ11a].

Considérons à présent une séquence de 2300 images, dans laquelle les images 1 et 2 ont été extraites. La caméra est en translation longitudinale et la scène est composée de 4 plans latéraux dans 500 images de cette séquence. Le taux de détection des paraboles est de **87%** durant cette phase.

L'analyse de la robustesse de l'algorithme a été détaillée dans [BZ11b]. Nous avons étudié de manière précise les effets de différentes perturbations pouvant affecter le processus de détection des surfaces planes. Les cinq sources d'imprécisions ou d'erreur ci-dessous ont été considérées une à une :

- Les approximations numériques dues notamment aux différentes discrétisations réalisées.

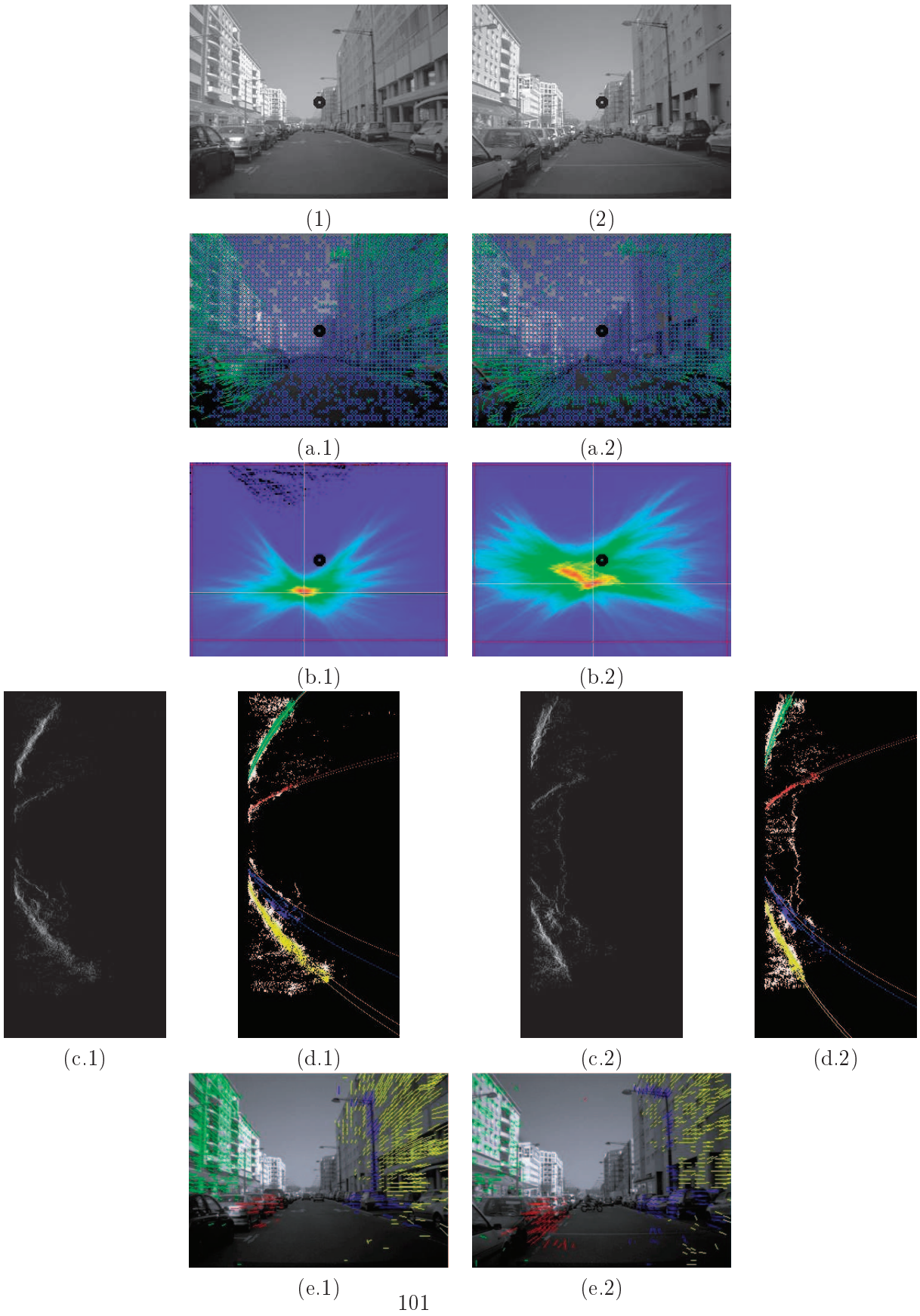


FIGURE 5.20 – Quelques résultats typiques. Image (1) correspond au cas où la caméra est en translation rectiligne. Image (2) correspond au cas où une moto traverse la rue. Dans l'image (3), le véhicule s'arrête. Pour chaque image, les images (b) donnent le résultat de la détection du FoE, les images (c) l'espace *c-vélocité* "bâtiment" , les images (d) l'espace *c-vélocité* après K-means clustering et les images (e) la détection de plans latéraux finale.

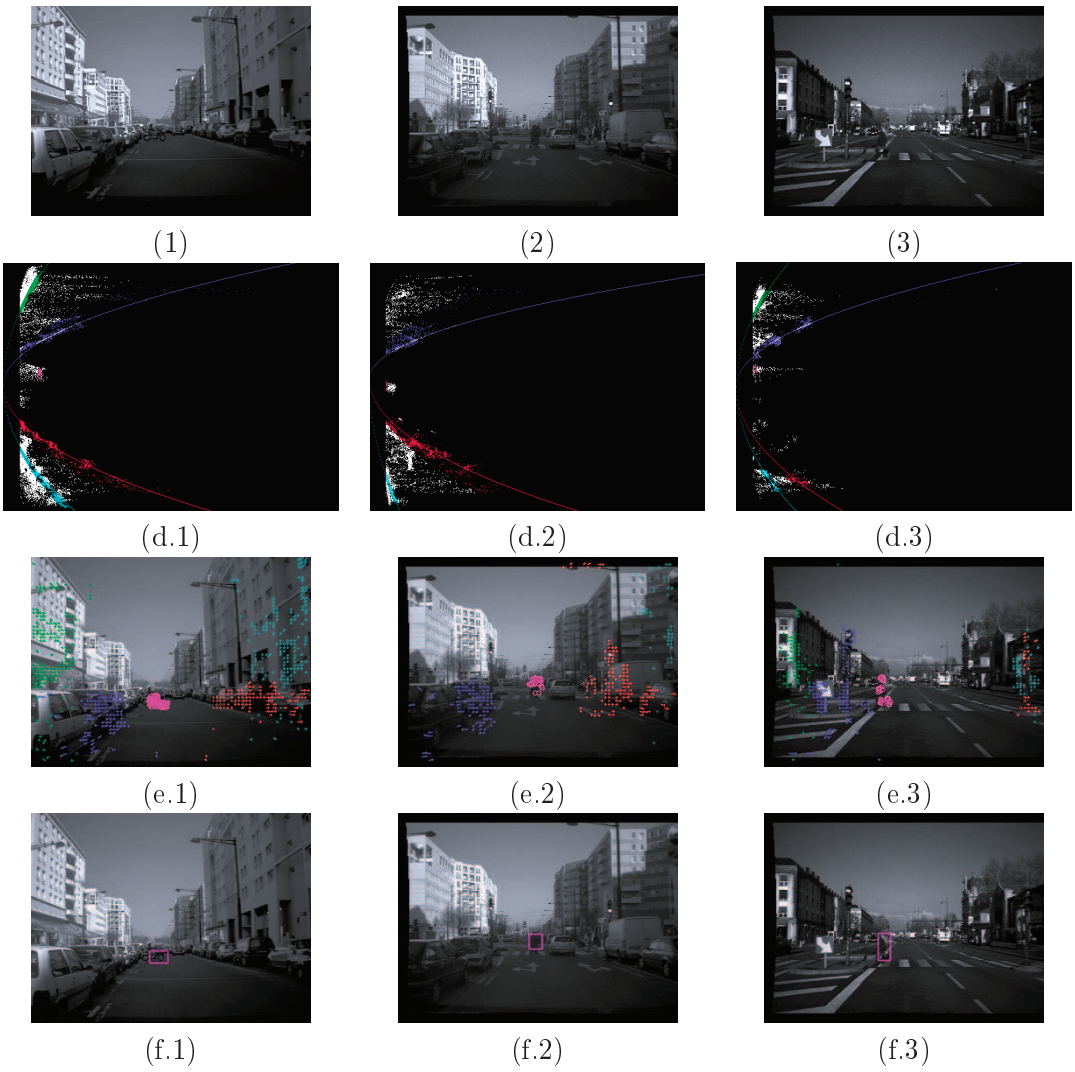


FIGURE 5.21 – Les images (1), (2) and (3) sont extraites d'une séquence de 1000 images. (d.1), (d.2) et (d.3) sont les espaces c-vélocité associés au modèle bâtiment. Les plans verticaux sont étiquetés. Les plans dans l'image correspondants sont affichés dans les images (e.1), (e.2) and (e.3). Les obstacles sont détectés (rectangles englobants) par analyse conjointe des trois espaces c-vélocité bâtiment, route et obstacle approchant/traversant dans les images (f.1), (f.2) et (f.3)

- Un flot optique en entrée bruité.
- Un axe optique de caméra non nécessairement parallèle au plan de la route.
- Une erreur dans l'estimation de la position du FoE
- Les contaminations diverses des votes de pixels appartenant à d'autres types de plans que ceux considérés (contamination inter-modèles).

L'étude qualitative et quantitative des sources d'incertitude ainsi que les expérimentations réalisées ont confirmé la robustesse du procédé, dans la limite des hypothèses initiales posées. Mais il serait intéressant à présent de considérer des mouvements plus généraux incluant des rotations (virage, suspension des roues, etc.) ou des translations latérales (glissement des roues, etc.). Les premiers calculs effectués dans ce sens montrent que le procédé peut aisément être généralisé à condition de définir des espaces de vote en cascade dédiés chacun à une composante du mouvement. Ce travail mérite d'être approfondi et constitue un des volets de mes perspectives de recherche.

5.4.5 Odométrie visuelle par *c-vélocité* inverse, état des travaux en cours

Nous avons montré, dans la section précédente, comment, connaissant le FoE, les plans 3D de la scène sont représentés dans l'espace *c-vélocité* par des paraboles. Par ailleurs, les paraboles obtenues dans les espaces de vote adéquats seront d'autant plus fines que l'erreur sur l'estimation du FoE est faible [BZ11b]. Il nous a paru alors intéressant, dans le cadre de la thèse de **Adrien Bak**, d'exploiter ce biais pour estimer la position même du FoE. En effet, nous avons expliqué dans la section précédente comment à partir de la connaissance du FoE, une détection des surfaces planaires était possible à travers l'approche *c-vélocité*. Nous montrons dans cette section comment la connaissance d'un plan peut permettre inversement de détecter le FoE. Nous appellerons ce procédé *c-vélocité* inverse. Pour cela, Il est nécessaire non seulement de pouvoir isoler de l'image les différents plans mais aussi de quantifier l'aspect parabolique des représentations de ces plans, en exhibant une métrique reflétant la distance séparant un FoE *supposé* du FoE réel. Détaillons ci-dessous le procédé.

Définition d'une métrique adaptée pour la localisation du FoE : Supposons dans un premier temps que l'image contient au moins un plan. Ce plan, considéré par exemple comme un plan de type "bâtiment", est noté Π. La dispersion de la représentation d'un plan dans l'espace de vote peut être formalisée de la manière suivante :

$$\text{dispersion}_{(FoE)}(\Pi) = \sum_{m \in \Pi} \left(c_{observée} - c_{moyenne}(w(m))\right)^2 \tag{5.14}$$

Dans cette équation, $c_{observée}$ est la $c-value$ du point m et $c_{moyenne}(w(m))$ est la moyenne de $c-values$ des points pour lesquels la norme du flot optique est identique à celle de m. En d'autre termes, cette dispersion est la somme des largeurs élémentaires de la courbe représentant Π ; ces largeurs étant considérées comme les carrés des écarts à la moyenne locale. Il nous a paru intéressant d'utiliser cette dispersion comme une métrique, monotone avec la distance séparant le FoE *supposé* du FoE *réel*. Elle est de plus convexe et peut permettre l'utilisation de techniques d'optimisation classiques afin de retrouver la

position du FoE. Dans le but d'exprimer la dispersion dans l'espace de vote, nous choisissons de considérer le lieu des points à w constant dans l'image. Ce lieu, noté $\mathcal{C}$, est en effet fixe, quelle que soit l'hypothèse de FoE considérée. Nous exprimons ensuite les $c-values$ associées aux points de ce lieu. Ainsi :

$$c^2_{batiment}\left(m \middle| \begin{matrix} x \\ y \end{matrix} \in \mathcal{C}\right) = K^2 - \Delta x x^2(2x - 2x_{FoE} - \Delta x) + \\ \Delta y^2 x^2 \mp \Delta y\sqrt{x^2K^2 - x^4(x - x_{FoE})} \tag{5.15}$$

Dans cette équation, K^2 est une constante, qui correspond à la $c-value$ *théorique* à laquelle on aboutit en menant les calculs avec un FoE *supposé* coïncidant avec le FoE *réel*. Le décalage du FoE *supposé*, par rapport au FoE *réel* est exprimé par Δx et Δy, x_{FoE} correspond à la coordonnée du *vrai* FoE. Il apparait que les variations de $c_{batiment}$, donc la dispersion, vont directement être croissantes en fonctions des deux composantes du décalage du FoE, Δx et Δy. La mesure de la dispersion, exhibée dans l'équation (5.14) est ainsi utilisée comme prévu en tant que mesure reflétant la qualité d'un FoE supposé et rappelons que cette métrique est convexe.

A partir de cette mesure, il est possible de formaliser le problème de localisation du FoE comme un problème de moindres carrés :

$$\epsilon^2 = \sum_{\Pi \in \mathcal{P}} \text{dispersion}_{(FoE)}(\Pi) \tag{5.16}$$

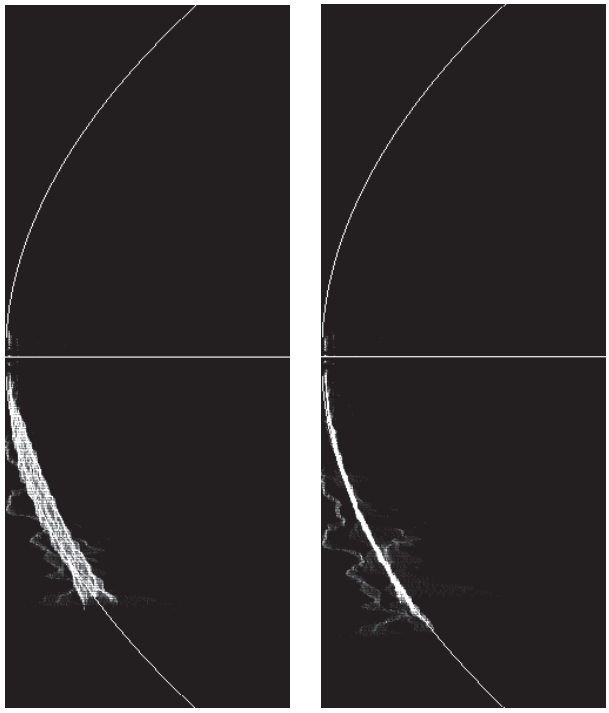
où $\mathcal{P}$ est l'ensemble des plans présents dans l'image. Un tel problème peut être résolu par un schéma d'optimisation classique. Nous avons choisi d'utiliser une descente de gradient.

Estimation de la Structure : L'approche dans sa première version nécessite une extraction préalable de plans verticaux et horizontaux. Dans nos premières expériences nous avons opté pour l'extraction des plans horizontaux, l'utilisation de la *v-disparité*. En ce qui concerne les plans verticaux, nous avons développé une transformée de Hough généralisée, très proche de celle présentée par [IKB01]. Dans cette transformation, nous utilisons un espace de vote à deux dimensions $\mathcal{V} = \{\rho, \theta\}$ où ρ représente la distance du plan à l'origine, et θ l'angle entre l'axe optique et le plan. A partir de cet espace de vote, il est possible d'extraire de l'image les points appartenant à des plans verticaux, satisfaisant aux hypothèses "bâtiment" et "obstacle". La combinaison de ces deux processus nous permet donc d'identifier dans l'image les différents plans d'intérêt, comme sur la Figure 5.23.

5.4.5.1 Résultats

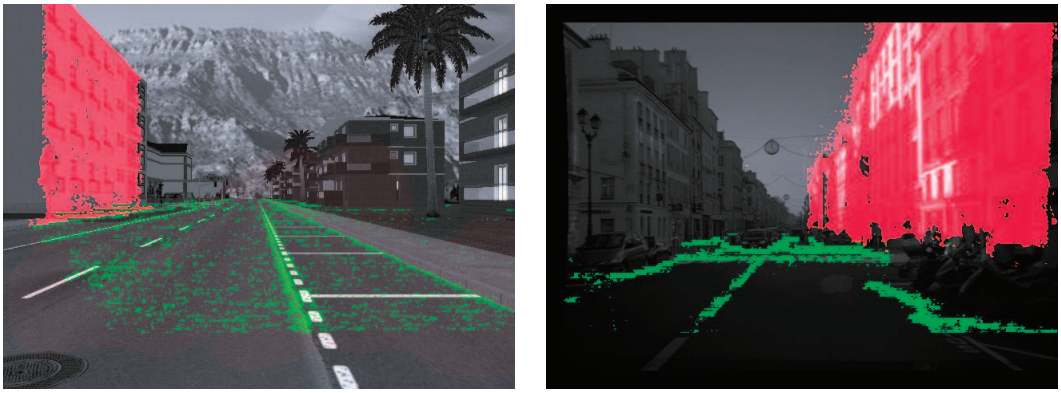
Images de Synthèse : Dans un premier temps, l'approche a été testée sur des images synthétiques. Nous avons étudié sa sensibilité à plusieurs facteurs : le bruit sur le flot optique, l'influence des rotations du capteur ainsi que la variation du nombre de plans utilisés. Ces premiers tests ont permis d'aboutir aux conclusions suivantes [BBA11b] :

- Sur un flot optique synthétique, et en l'absence de bruit, une extraction exacte de la position du FoE peut être correctement réalisée et ce, quelle que soit sa position.



(a) (b)

FIGURE 5.22 – Visualisation de l'espace de vote "bâtiment" avant et après optimisation. La déformation de la représentation *c-vélocité*, induite par la position du FoE avant optimisation est visiblement compensée.



(a) (b)

FIGURE 5.23 – Résultats du Système d'Estimation Structurelle. (a) : Images Simulées. (b) : Images Réelles

- Après ajout de divers types de bruits sur le flot synthétique, il apparaît que la localisation du FoE est très robuste à ces perturbations.
- Les taux de rotation importants (au dessus de $0,05\text{rad}/frame$) introduisent une erreur assez importante. Pour des rotations plus faibles, de l'ordre de grandeur de celles qui peuvent apparaître lors d'une trajectoire en "ligne droite", cette erreur est beaucoup plus contenue.
- Nous n'avons pas noté d'influence notable du nombre de plans utilisés. Il reste à vérifier si la nature des plans (route, bâtiment, obstacle) joue un rôle important.

Images Pseudo-Réalistes : Cette méthode a été testée sur les images pseudo-réalistes générées par le simulateur SiVIC [11] Le simulateur a synthétisé une séquence de 250 paires stéréo, prenant place dans un environnement urbain au trafic modéré. Dans cette séquence, le mouvement du véhicule est majoritairement translationnel. Un exemple d'extraction du FoE est donné dans la FIGURE 5.24. On peux constater que le FoE extrait repose sur la ligne d'horizon, ce qui est cohérent avec le mouvement connu du mobile. Pour cette image, l'erreur commise entre le FoE extrait et le vrai FoE, recalculé à partir des composantes du mouvement, est de $2,2$ pixels. Sur la totalité de la séquence considérée, l'erreur moyenne commise est de $5,2$ pixels, avec une erreur maximale de $15,6$ pixels. Cette erreur peut paraître importante dans l'absolu, il nous semble toutefois pertinent de la relativiser : elle est exprimée en pixels, or le FoE ne doit pas être traité au même titre qu'un simple point d'intérêt, il représente avant tout une mesure des différentes translations d'un véhicule. En effet, si nous considérons le système optique simulé (qui présente une longueur focale de $f = 10mm$ et une taille de pixel de $10\mu m$), une erreur de 1 pixel sur la position du FoE se traduit par une erreur de 10^{-3} sur le rapport :

$$\frac{\|T\|}{T_Z} \tag{5.17}$$

où $T = \begin{vmatrix} T_X \\ T_Y \end{vmatrix}$. Cela représente, dans le cas d'un véhicule se déplaçant à $50km/h$, une translation latérale de $0,6mm$. Dans notre cas, l'erreur commise sur la localisation du FoE correspond donc à une erreur d'estimation du rapport $\frac{\|T\|}{T_Z}$ de 0.8%. A titre de comparaison, nous avons également utilisé une méthode de localisation par vote cumulatif, telle qu'illustrée dans [SJBK08]. Cette dernière méthode conduit à une erreur moyenne de localisation du FoE de $10,6$ pixels, soit une erreur de $1,6\%$ sur l'estimation du rapport $\frac{\|T\|}{T_Z}$.

Images Réelles : L'approche a été testée sur les images issues des bases de données réalisées dans le cadre du projet LoVE. Le flot optique utilisé a été calculé en utilisant la méthode FOLKI [BC05]. Malgré les moyens expérimentaux mis en oeuvre, aussi bien par nous-même que par les différentes équipes qui nous ont autorisé l'accès à leurs bases de données, il n'a pas été possible d'utiliser de capteurs suffisamment précis pour fournir une vérité terrain exploitable comme base de comparaison. Au mieux, en utilisant les données de l'Université de Karlsruhe nous pouvons espérer une estimation de la position du FoE

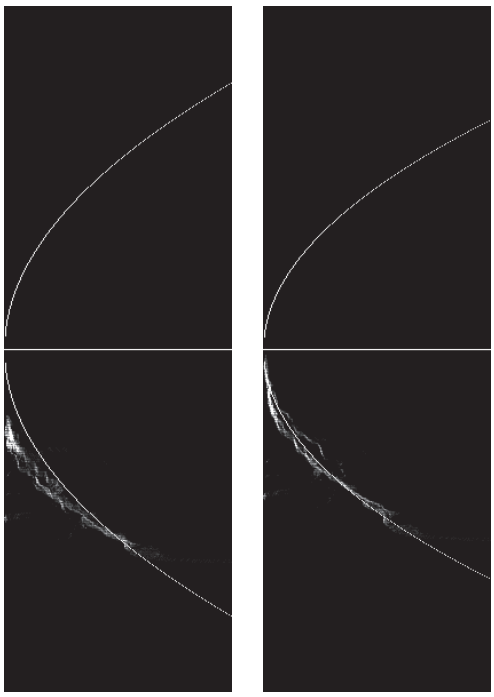
11. Commercialisé par le LIVIC (IFSTTAR).

FIGURE 5.24 – Image issue d'une séquence simulée, le flot optique est calculé par la méthode FOLKI et le FoE par la *c-vélocité* inverse

précise à environ 13 pixels près, ce qui n'est pas suffisant pour pouvoir constituer une base solide de comparaison. L'évaluation de la qualité de l'extraction du FoE par notre approche s'est donc faite de manière qualitative sur les images réelles. Ainsi, les images de la FIGURE 5.25 montrent le Foyer d'Expansion extrait par notre méthode pour plusieurs images réelles. Dans tous les cas, ce FoE correspond à la connaissance que nous avons du mouvement approximatif de l'ego-véhicule. De plus, il peut être intéressant de considérer les lignes de champ du flot optique comme des indicateurs visuels de la qualité du FoE extrait.

FIGURE 5.25 – Exemple de résultats obtenus sur image réelle

Un autre indicateur qualitatif de la qualité du FoE extrait peut être la comparaison entre l'espace de vote initial (calculé avec un FoE *supposé* au centre de l'image) et l'espace de vote final (calculé avec le FoE *extrait*). Une telle comparaison est illustrée dans les FIGURES 5.22 et 5.26. Dans les deux cas, il apparaît que, à l'issue de l'optimisation réalisée, la représentation *c-vélocité* des plans observés est bien plus conforme au modèle parabolique attendu, signe que la position estimée du FoE est plus proche du FoE réel après optimisation qu'avant.



(a) (b)

FIGURE 5.26 – Comparaison des Espaces de vote avant (a) et après (b) recherche du FoE

5.4.6 Conclusion sur la décision cumulative 2D en cascade

La décision cumulative en cascade présentée dans cette section se décline en deux techniques à finalités différentes mais pouvant cependant être combinées et employées conjointement.

- La première approche fournit une détection de plans 3D par cumul de vecteurs vitesse 2D. Deux espaces de vote sont traités de manière séquentielle : l'espace *c-vélocité* et l'espace de Hough 1D découlant de la paramétrisation des paraboles. Les perspectives immédiates apparaissent : la généralisation à des surfaces paramétrées plus complexes qui augmentera probablement la dimension de l'espace de Hough initialement 1D et la prise en compte de mouvements plus généraux incluant les deux autres translations ainsi que les rotations. Un premier examen des équations du mouvement initiales suggère que l'espace c-vélocité sera lui même décomposé en plusieurs espaces en cascade chacun associé à un paramètre du mouvement. Ceci fera l'objet à court terme d'une étude détaillée.
- La seconde méthode propose un moyen d'estimer le FoE à travers le procédé de *c-vélocité* inverse. Contrairement à la majorité des méthodes existantes, elle n'utilise pour cela qu'une partie de l'information contenue dans le flot optique, en l'occurrence, la norme relative. De ce fait, l'approche est insensible au bruit sur l'orientation des vecteurs et au biais sur leur estimation. Par ailleurs, outre le fait que cela constitue en soi une tentative d'estimer l'ego-mouvement translationnel, cette approche permet-

tra, une fois combinée à la *c-vélocité* directe, un re-bouclage pour une estimation de plus en plus précise de la nécessaire position du FoE (thèse en cours de **Nie Qiong**). Ce travail une fois achevé permettra une estimation simultanée de la structure de la scène et de l'ego-mouvement translationnel conforme au point de vue que nous défendons sur les futurs systèmes monoculaires, efficaces et compacts.

5.5 Conclusion

Dans ce chapitre, nous avons montré comment un choix de primitives robustes associé à un processus de décision cumulatif permettait la réutilisabilité des opérateurs dans tous les secteurs. Les systèmes proposés ont la particularité d'être compacts et cohérents, propriétés recherchées dans les applications considérées. Nous avons pu traiter différentes phases de la perception dynamique : le cas où la caméra est fixe et les objets mobiles, le cas où la caméra est en mouvement avec connaissance a priori sur le modèle de ce mouvement et le cas plus général de l'estimation de la structure à partir du mouvement. Chacune des alternatives traitées ouvre une multitude d'extensions et de perspectives à court, moyen et long terme dans lesquelles nous comptons nous impliquer activement.

Perspectives et conclusion

Mes travaux de recherche portent essentiellement sur l'analyse de scènes à partir de caméras mobiles avec pour application immédiate l'apport d'une vision par ordinateur efficace dans les systèmes d'aide à la conduite. L'idée initiale est que l'autonomie d'un système implique, ne serait-ce que pour raisons énergétiques, une faible variété d'opérateurs de perception, dont les algorithmes de vision. Nous avons montré que cela était possible à travers le choix de primitives image adaptées et d'un processus de décision manipulant un objet unique (l'histogramme). D'autant plus que les exigences des applications liées aux systèmes d'aide à la conduite en terme de robustesse et de fiabilité (la sécurité du conducteur est en jeu) justifient encore les techniques cumulatives, notre approche assurant ainsi une décision fiable. Les travaux réalisés nous ouvrent alors trois perspectives de recherche, trouvant une application pertinente dans les projets et collaborations dans lesquels je suis actuellement impliquée et sur lesquels je compte m'investir.

Le premier volet de mes perspectives de recherche vise à tirer profit au maximum de la vision monoculaire dans les tâches d'estimation du mouvement propre du véhicule et de la structure de la scène. En effet, les approches existantes tant au niveau national qu'au niveau international se sont pour la plupart focalisées sur d'autre types de capteurs que la vision (capteurs extéroceptifs tels que les Radar ou Lidar et proprioceptifs tels que les odomètres, accéléromètres ou gyromètres). Lorsque la modalité vision est choisie, elle est cependant systématiquement "dupliquée" (i.e. stéréovision) assurant ainsi une estimation de la structure de la scène confortable car bien maitrisée lorsque les caméras sont correctement calibrées. Notre point de vue rejoint les efforts qui commencent à émerger dans ce domaine concernant l'utilisation de la vision monoculaire. En effet, ajouté au fait que les techniques existantes peinent à être industrialisées en raison du coût engendré par la multiplicité des capteurs, il est évident que ces approches ne pourraient que tirer profit de l'exploitation des informations provenant des séquences monoculaire. Pour ne donner qu'un exemple : la stéréovision seule ne permet en aucun cas d'estimer le mouvement des obstacles (s'approchent-ils ou pas du véhicule ?). En revanche, l'analyse des séquence monoculaires, en plus de pouvoir estimer ce mouvement est tout à fait à même de fournir des informations sur la structure de la scène. C'est dans cet esprit que nos efforts de recherche se sont concentrés et sont confortés par une tendance générale parmi les chercheurs du domaine. Pour cela, nous avons montré que si l'on contraignait la nature du mouvement caméra (ex. translation) et les objets (approximation planaire), il est possible d'identifier les deux simultanément. Les premiers travaux entrepris dans ce sens nous ouvrent des perspectives très prometteuses. Celles-ci se focaliserons alors sur trois aspects principaux :

- La généralisation de l'approche *c-vélocité* que nous avons proposée à d'autres types de mouvement du capteur. En effet, bien que le déplacement du véhicule soit majoritairement longitudinal, des translations non négligeables peuvent être relevées en raison des suspensions et des déviations possibles pendant les virages. Par ailleurs, les rotations dues aux virages eux mêmes, les élévations produites par les variations de la pente de la route et le tangage inévitable sur les virages, nous obligent à introduire ces transformations dans le modèle du mouvement propre du capteur. De la même manière, l'hypothèse que la scène peut être approximée par un ensemble de plans (la route, les bâtiments en environnement urbain, les piétons) a ses limites.

Les routes ne sont généralement pas planes et encore moins les obstacles de type "piéton". Nous devons alors généraliser notre approche pour qu'elle tienne compte de surfaces paramétrées non nécessairement planaires. Dans les deux cas, les vitesses 3D puis 2D obtenues par projection dans l'image des surfaces paramétrées doivent être exprimées en fonction des 6 degrés de liberté conduisant ainsi à de nouvelles courbes d'iso-vitesse dans l'images et à de nouveaux espaces de votes probablement uni-dimensionnels mais en cascade. Les aspects implémentation efficace de ces espaces de vote en cascade seront étudiés précisément.

- La mise en oeuvre d'une réelle "coopération" entre les divers espaces de vote construits. En effet, l'une des originalités de notre approche est de s'affranchir de l'estimation précise des orientations des surfaces planaires en discrétisant l'espace des orientations possibles et en l'instanciant par des espaces de vote dédiés. Cela a cependant deux inconvénients majeurs si l'on ne pousse pas plus loin l'analyse : d'abord, le choix de la discrétisation conditionne la précision sur la mesure de l'orientation. Par ailleurs, les pixels quelle que soit l'orientation du plan auquel ils appartiennent votent dans tous les espaces même ceux ne correspondant pas à leur modèle d'orientation, conduisant ainsi à perturber les structures devant émerger. Concernant le premier point, nous envisageons d'utiliser une discrétisation itérative progressive inspirée des approches multi-résolution. Nous passerons d'un découpage grossier de l'espace des orientations à un découpage plus fin de manière itérative en fonction des structures qui émergent provisoirement dans les espaces de vote. Concernant le deuxième point, nous comptons nous inspirer de techniques de segmentation d'images multi-canaux basés sur l'histogramme [OPR78]. Cette segmentation itérative permet de localiser les pics dans un espace de vote à une itération donnée tout en supprimant la contribution des pixels ayant engendré ce pic à l'itération suivante. Appliquée aux "histogramme" de type *c-velocité*, elle permettrait de tenir compte des contributions au vote de plusieurs modèles d'objets à la fois, tout en garantissant que chacun d'eux puisse émerger dans l'espace de vote qui lui est dédié.

- L'étude précise des différentes sources d'erreur pouvant entacher le processus de décision basé sur la *c-velocité*. Nous en avons dénombré au moins 5 : les erreurs sur le modèle d'objet (orientations supposées des surfaces), sur le modèle de mouvement si celui-ci est simplifié, sur la localisation du Foyer d'expansion (origine du repère 2D image permettant le calcul des iso-courbes), par la contamination inter-modèles dans les différents espaces de vote que nous avons présentés ci-dessus et sur l'estimation des vecteurs vitesse. Concernant ce dernier point, nous avons d'ores et déjà entrepris une étude de la précision des différentes techniques d'estimation du flot optique (thèse de Q. Nie), incluant aussi bien les techniques denses que les techniques par mise en correspondance de primitives éparses.

Le deuxième volet a pour ambition de s'intéresser à l'accumulation d'évidence non plus au sein d'un seul processus visuel mais en faisant coopérer plusieurs modalités distinctes de la vision. Notre point de vue rejoint celui des chercheurs travaillant sur les thèmes liées à la fusion de données multi-capteurs ou aux systèmes coopératifs. En effet, les recherches dans

ce domaine sont en pleine expansion et motivées par plusieurs facteurs : d'abord chacun des capteurs fournit des mesures pouvant être entachées d'erreur, ensuite leur défaillance ponctuelle peut conduire à des données manquantes, enfin la disponibilité des données à un instant t peut varier d'un capteur à l'autre. L'idée la plus immédiate est donc de faire coopérer plusieurs capteurs afin de renforcer les décisions. Nous proposons de rester dans cet esprit de renforcement (donc d'accumulation) mais au sein d'un même capteur "caméra". En effet, nous sommes persuadée que toutes les possibilités de la vision ne sont pas encore exploitées pleinement. La coopération mouvement/stéréovision en est un exemple. Un obstacle sera mieux détecté conjointement par stéréovision type *v-disparité* et par déplacement type *c-velocité*. Nous pouvons ainsi définir plusieurs types de "capteurs logiques" où chacun d'eux se chargerait d'un processus de vision précis.

Le troisième volet concerne l'optimisation et la parallélisation des algorithmes proposés. Ils posent en effet dans leur ensemble des problèmes intéressants quant à l'adéquation algorithme/architecture. La formalisation nécessaire à une implantation systématique devrait bénéficier d'une définition fonctionnelle des nouveaux opérateurs proposés. On notera que dans notre recherche, formellement, du plus bas au plus haut niveau sémantique on passe de la recherche de déplacement d'ensembles de niveau constant à la recherche d'ensembles de déplacements constants des niveaux, au prix d'une contrainte ajoutée sur la forme de ces derniers : ce même vocabulaire dans un ordre différent n'exprime rien d'autre que le maintien des mêmes opérateurs contrôlés différemment, minimalisme encore mieux traduit par l'expression fonctionnelle :

$$\partial_t \left(1_{n=c^{ste}}\right) \cong c^{ste} \; vs. \; 1_{\partial_t(n)=c^{ste}} \cong c^{ste}$$

Par ailleurs, il est clair que la complexité engendrée par la généralisation de nos approches *c-vélocité* devra être étudiée précisément. La gestion des espaces de vote creux et en cascade en est un exemple intéressant.

Projets de recherche

Mes perspectives de recherche seront mises en oeuvre sur des systèmes et validées dans le cadre de trois projets ou collaborations à l'horizon des cinq années à venir. Ces projets sont décrits ci-dessous.

Détection d'obstacles à partir de caméras mobiles

Le premier projet s'inscrit dans le cadre d'une colloboration avec le LIVIC ayant pour thème principal la validation et l'expérimentation en conditions réelles des approches basées sur la coopération mouvement/stéréovision pour la détection d'obstacles. L'approche mise en oeuvre a été pensée de telle manière que chaque élément de la chaîne puisse s'appuyer sur les travaux scientifiques actuels du domaine de la vision par ordinateur les plus performants et les plus adaptés à ce type d'applications. Pour cela, nous avons bénéficié aussi bien de l'expertise du LIVIC dans ce domaine que de l'expérience et des travaux de recherche menés jusque là à l'IEF. L'utilisation conjointe des informations provenant du

mouvement et de la stéréovision nous a permis d'obtenir des résultats très prometteurs, notamment dans des situations classiquement difficiles à traiter : mouvements lents, occultations partielles des objets. Les résultats obtenus par l'utilisation exclusive de capteurs de vision sont comparables à, voire meilleurs que, beaucoup d'approches nécessitant des capteurs supplémentaires (centrales inertielles). A ce jour, notre objectif est la mise en oeuvre effective de l'approche proposée sur des véhicules équipés afin de réaliser une évaluation rigoureuse de l'approche. Je pourrai bénéficier pour cela des infrastructures du LIVIC [12] afin de mener convenablement ce projet. Ces moyens mis à ma disposition me seront indispensables pour mener à bien une évaluation où les variables expérimentales (variabilité des conditions d'acquisition : différentes instants de la journée, sous diverses conditions climatiques, avec divers niveaux de visibilité et d'éclairement) sont de première importance en traitement des images, en particulier pour des applications liées aux aides à la conduite où la sécurité du conducteur est en jeu.

Scenarii d'automatisation à basse vitesse (*platooning*)

Les algorithmes coopération mouvement/stéréovision mis en oeuvre pourraient permettre d'améliorer les scénarii d'automatisation à basse vitesse traités au LIVIC. En particulier, nous avons ciblé une nouvelle technique baptisée *platooning* (ou convoi de véhicules) permettant aux voitures de rouler plus près les unes des autres afin de réduire les émissions de dioxyde de carbone et d'économiser de l'essence. La réalisation du *platooning* nécessite le suivi d'une trajectoire de référence apprise, ou bien le suivi de la trajectoire du "leader" du convoi (véhicule situé en tête du convoi). Elle nécessite d'estimer le mouvement propre de chaque véhicule ainsi que les distances inter-véhicules. Pour ces trois briques de bases : suivi, ego-mouvement et inter-distances, nos travaux restent à être validés et appliqués au cas particulier du *platooning*.

Vidéo-surveillance à partir d'un véhicule de contrôle mobile

Le projet Européen SPY [13] (Surveillance imProved sYstem) a pour but de réaliser de nouveaux systèmes d'assistance et de surveillance, intelligents et automatisés, adaptés pour la mobilité des utilisateurs finaux sur le terrain. Les utilisateurs potentiellement intéressés sont les forces de sécurité (policiers), les pompiers ou les autorités locales. Ce projet s'inscrit parmi les nouvelles générations de projets liés à la vidéo-surveillance. En effet, alors que pendant de nombreuses années, le vidéo-surveillance se basait exclusivement sur des caméras fixes dans des zones protégées bien définies, elle tente de s'étendre actuellement à des cas plus généraux où les environnement sont complexes, fortement dynamiques et non nécessairement connus et où les capteurs sont embarqués sur des véhicules mobiles assurant la surveillance. Cette application pose des problèmes similaires en terme d'analyse

12. En effet, le LIVIC est le laboratoire français de référence dans le domaine des systèmes d'aide à la conduite : il dispose d'un réseau de 7km de pistes d'essais, d'une flotte de 4 véhicules instrumentés, de plus de 600 mètres carrés d'ateliers et laboratoires pour le développement des prototypes, de bancs de test et d'étalonnage, de moyens de mesure et de recueil de données et d'une salle de simulation.

13. Ce projet s'inscrit dans le programme européen ITEA2 et a obtenu son label le 9 décembre 2009. Les partenaires du projet sont : IEF, ENSTA, PPSL, Cassidian, ARCINFO, IT sud Paris, EOLANE (France), COGVIS (Australie), Multitel (Belgique), VTT, KILOSOFT (Finlinde), Aselsan, C2TECH (Turquie).

d'images à celles évoquées pour les aides à la conduite. Ma contribution dans ce projet concerne l'estimation du mouvement et le suivi des objets mobiles après compensation du mouvement propre du capteur. Elle devrait conduire à fournir une sorte de carte obtenue après étiquetage des régions de l'image en fonction de leur mouvement et de leur structure approchée (exemple : bâtiment vertical situé à une distance d, obstacle frontal ayant un mouvement translationnel, obstacle "fuyant", voiture garée, voiture s'approchant sur la file opposée, etc.).

Réalité augmentée spatiale sur supports multi-plans

Dans le cadre d'un rapprochement entre l'IEF et le LIMSI, j'ai entrepris une nouvelle collaboration avec l'équipe AMI (Architectures et Modèles pour l'Interaction) à travers le montage de deux projets. Le premier, d'une durée de deux ans, démarrant en 2011, est une Action Incitative financée par le LIMSI et l'IEF et le second est une ANR (Virtual Palimpsest [14]), devant être soumise en mai 2012, toutes deux portant sur le concept de Réalité Augmentée Spatiale qui consiste à superposer de l'information numérique sur le monde physique à l'aide de projecteurs. Deux types de projections nous intéressent : le premier scénario concerne les scènes d'intérieur (salle de réunion, pièce de bureau, salle de musée, etc) et le second les scènes en extérieur (bâtiments). Il est supposé que la scène est composée de plusieurs surfaces planaires exploitables dans le cadre d'une projection. La contribution de l'équipe VISA concerne la recherche d'une surface montrant le maximum de similarité avec une surface de type écran. Il s'agit de trouver la surface répondant au mieux à une combinaison de critères géométriques (privilégier la planéité tout en obtenant une distance surface/système permettant une visualisation de qualité et respecter les contraintes liées à la technologie du vidéoprojecteur), colorimétriques (privilégier des surfaces aux couleurs compensables par le système), texturaux (privilégier des surfaces de faible rugosité et homogènes en couleur) et morphologiques (surfaces de forme rectangulaire).

Conclusion

L'analyse du mouvement constitue un champ de recherche passionnant dans lequel je me suis investie depuis plusieurs années et qui est devenu mon domaine d'expertise. Ce mémoire a tenté de retracer la cohérence de mon parcours de recherche et d'enseignement. Les moyens humains limités ne m'ont pas empêché d'aboutir à des résultats intéressants grâce à un investissement appréciable. Je suis persuadée de continuer à contribuer au développement des connaissances et des applications en vision dynamique, notamment avec une équipe qui se regrouperait sur le sujet.

14. les partenaires de cette ANR sont à ce jour : le LIMSI-CNRS (AMI), le LIG (IIHM), le MAP GAMSAU, la société Immersion, l'IGN (laboratoire MATIS) et l'IEF (VISA).

Bibliographie

[Alo90] J. Y. Aloimonos. Perspective approximations. *Image and vision computing*, 8(3) :177–192, 1990.

[Ana01] P. Anandan. Video registration and motion estimation : a retrospective. In *Conference on Recent Advances in 3D Digital Imaging and Modeling*, june 2001.

[Bai85] H.S. Baird. *Model-based image matching using location.* 1985.

[Bal81] D. H. Ballard. Generalizing the hough transform to detect arbitrary shpaes. *Pattern recognition journal*, 13(2) :111–122, 1981.

[BBA10] A. Bak, S. Bouchafa, and D. Aubert. Detection of independent-moving objects through stereo-vision and ego-motion extract. In *IEEE Intelligent Vehicules Symposium*, pages 863–870, San Diego, CA, June 21–24 2010.

[BBA11a] A. Bak, S. Bouchafa, and D. Aubert. Dynamic object detection through visual odometry and stereovision. *Machine Vision and applications, special issue on car navigation and vehicule systems*, 2011.

[BBA11b] A. Bak, S. Bouchafa, and D. Aubert. Focus of expansion localization through inverse c-velocity. In *International Conference on Image Analysis and Processing*, 2011.

[BBB+01] A. Bensrhair, M. Bertozzi, A. Broggi, P. Miche, S. Mousser, and G. Toulminet. A cooperative approach to vision-based vehicle detection. *IEEE Intelligent Transportation Systems*, pages 207–212, 2001.

[BC05] G. Le Besnerais and F. Champagnat. Dense optical flow by iteratuve local window registration. In *IEEE Inter. Conf. On Image processing ICIP*, 2005.

[BD98] S. D. Buluswar and B. A. Draper. Color machine vision for autonomous vehicles. *International Journal for Engineering Applications of Artificial Intelligence*, 11(2) :245–256, 1998.

[BDW06] T. Bailey and H. Durrant-Whyte. Simultaneous localization and mapping (slam). *Part II. Robotics & Automation Magazine*, 13(3) :108–117, 2006.

[BETG08] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Surf : Speeded up robust features. *Computer Vision and Image Understanding (CVIU)*, 110(3) :346–359, 2008.

[Bou98] S. Bouchafa. *Détection du mouvement insensible aux variations de contraste. Application à la détection de comportements de foules anormaux dans le métro par traitement d'images.* PhD thesis, Université Paris VI, Dec. 1998.

[BPU+08] C. Braillon, C. Pradalier, K. Usher, J. Crowley, and C. Laugier. Occupancy grids from stereo and optical flow data. *Experimental Robotics.*, 39 :367–376, 2008.

[Bro63] R. G. Brown. *Smoothing Forecasting and Prediction of Discrete Time Series.* Englewood Cliffs, NJ : Prentice-Hall, 1963.

[Bro92] L.G. Brown. A survey of image registration techniques. *ACM computing surveys*, 24(4) :325–376, 1992.

[BS72] D.I. Barnea and H.F. Silverman. A class of algorithms for fast digital image registration. *IEEE Trans. on Computers*, 21(2) :179–186, Feb. 1972.

[BTG06] H. Bay, T. Tuytelaars, and L. Van Gool. Surf : Speeded up robust features. In *European Conference on Computer Vision*, 2006.

[BZ06] S. Bouchafa and B. Zavidovique. Efficient cumulative matching for image registration. *Image and vision computing*, 24(1) :70–79, 2006.

[BZ11a] S. Bouchafa and B. Zavidovique. C-velocity : a flow-cumulating uncalibrated approach for 3d plane detection. *International Journal of Computer Vision*, 2011.

[BZ11b] S. Bouchafa and B. Zavidovique. Error sources and their influence on c-velocity methods. *Pattern recognition and Image Analysis*, 21(3), 2011.

[BZ11c] S. Bouchafa and B. Zavidovique. Robustness of c-velocity methods for 3d plane detection. *Pattern recognition and Image Analysis*, 21(2), 2011.

[CCM99] V. Caselles, B. Coll, and J. Morel. Topographic maps and local contrast change in natural images. *International Journal of Computer Vision*, 33 :5–27, 1999.

[CGPP03] R. Cucchiara, C. Grana, M. Piccardi, and A. Prati. Detecting moving objects, ghosts and shadows in video streams. *IEEE Trans. on Patt. Anal. and Machine Intelligence*, 25(10) :1337–1342, 2003.

[CMR10] A. I. Comport, E. Malis, and P. Rives. Real-time quadrifocal visual odometry. *International Journal of Robotic Research, Special Issue on Robotic Vision*, pages 486–492, 2010.

[Cou95] C. Coutelle. *Conception d'un système à base d'opérateurs de vision rapides.* PhD thesis, Université de Paris 11, Orsay, FRANCE, 1995.

[CT77] M. Cohen and G.T. Toussaint. On the detection of structures in noisy pictures. *Pattern recognition*, 9 :95–98, 1977.

[DC00] F. Dornaika and R. Chung. Cooperative stereo-motion : Matching and reconstruction. *Computer Vision and Image Understanding*, 79(3) :408–427, 2000.

[DH72] R.O. Duda and P. E. Hart. Use of the hough transformation to detect lines and curves in pictures. *Comm. ACM*, 15 :1115, January 1972.

[Dic02] E. Dickmanns. The development of machine vision for road vehicles in the last decade. In *IEEE Intelligent Vehicles Symposium*, 2002.

[DT05] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition*, 2005.

[EDHD02] A. Elgammal, R. Duraiswami, D. Harwood, and L.S. Davis. Background and foreground modeling using non-parametric kernel density estimation for visual surveillance. *Proc. of the IEEE*, 90 :1151–1163, 2002.

[FA95] C. Fermuller and Y. Aloimonos. Global rigidity constraints in image displacement fields. In *International Conference on Computer Vision*, pages 245–250, june 1995.

[FFKM02] D. Fedorov, L. M. G. Fonseca, C. Kenney, and B. S. Manjunath. Automatic registration and mosaicking system for remotely sensed imagery. In *SPIE 9th International Symposium on Remote Sensing*, 2002.

[FRBG05] U. Franke, C. Rabe, H. Badino, and S. Gehrig. 6d-vision : Fusion of stereo and motion for robust environment perception. *Lecture notes in computer science*, 3663 :216–223, 2005.

[Fro99] J. Froment. A compact and multiscale image model based on level sets. In *Scape Space Theories in Computer Vision*, pages 152–163. Lecture notes in Computer science 1682, 1999.

[FWH07] F.Wu, L. Wang, and Z. Y. Hu. Foe estimation : Can image measurement errors be totally. *Pattern recognition*, 40(7) :1971–1980, 2007.

[GB00] J. Gama and P. Brazdil. Cascade generalization. *Machine Learning*, 41 :315–343, 2000.

[GBA00] F. Guichard, S. Bouchafa, and D. Aubert. A change detector based on level sets. In *International Symposium on Mathematical Morphology*, Palo Alto, USA, june 2000.

[Ger87] G. Gerig. Linking image-space and accumulator space. In *ICCV*, pages 112–117, 1987.

[GI89] D. Gusfield and R.W. Irwing. *The stable marriage problem - Structure and Algorithms*. MIT Press, 1989.

[Gir80] M. Girard. Digital target tracking. In *NATO Group*, 1980.

[GS84] L.O. Gorman and A. C. Sanderson. The converging squares algorithm : and efficient method for locating peaks in multidimensions. *IEEE Trans. on PAMI*, 6(3) :280–288, 1984.

[GSP86] A. Goshtasby, G.C. Stockman, and C.V. Page. A region-based approach to digital image registration with subpixel accuracy. *IEEE Trans. Geosci. Remote sensing*, 24(3) :390–399, 1986.

[Gui86] Y. Le Guilloux. A matching algorithm for horizontal motion, application to tracking. In *IEEE Proc. 8th Inter. Conf. on Pattern Recognition*, 1986.

[Har95] R.I. Hartley. In defense of the 8-point algorithm. In *IEEE International Conference on Computer Vision*, pages 1064–1070, 1995.

[HCD04] B. Han, D. Comaniciu, and L. Davis. Sequential kernel density approximation through mode propagation : applications to background modeling. In *Proc. ACCV - Asian Conf. On Computer Vision*, 2004.

[Hei02] S. Heinrich. Fast obstacle detection using flow/depth constraint. In *Intelligent Vehicle Symposium*, 2002.

[Hil92] E. C. Hildreth. Recovering heading for visually-guided navigation. *Vision Research*, 32(6) :1177–1192, 1992.

[HKM08] Douglas Hanes, Julia Keller, and Gin McCollum. Motion parallax contribution to perception of self-motion and depth. *Biological Cybernetics*, 98(4) :273–293, 2008.

[HN10] C. Huang and R. Nevatia. High performance object detection by collaborative learning of joint ranking of granules features. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010.

[Hol04] C. C. Holt. Forecasting trends and seasonal by exponentially weighted averages. *International Journal of Forecasting*, 10(1) :5–10, 2004.

[HTFF05] T. Hastie, R. Tibshirani, J. Friedman, and J. Franklin. The elements of statistical learning : data mining, inference and prediction. *The Mathematical Intelligencer*, 27(2) :83–85, 2005.

[IKB01] L. Iocchi, K. Konolige, and M. Bajracharya. Visually realistic mapping of a planar environment with stereo. *Lecture notes in Control and Information science*, 271 :521–532, 2001.

[IRP97] M. Irani, B. Rousso, and S. Peleg. Recovery of egomotion using region alignment. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 19(3) :268–272, 1997.

[KG82] P. Kuhl and C. Giardina. Elliptic fourier features of a closed contour. *Computer Graphics and Image Processing*, 18 :236–258, 1982.

[KMV94] S. Khuller, S. G. Mitchell, and V. V. Vazirani. Online algorithms forweighted bipartite matching and stable marriages. *Theorical Computer Science*, 127(2) :255–267, 1994.

[KTS98] T. Kalinke, C. Tzomakas, and W.V. Seelen. A texture-based object detection and an adaptive model-based classification. In *IEEE Intelligent Vehicles Symposium*, 1998.

[LA99] H. Lester and S. R. Arridge. A survey of hierarchical nonlinear medical image registration. *Pattern Recognition. Special issue on Image Registration*, 32(1) :129–149, 1999.

[LAT02] R. Labayrade, D. Aubert, and J.P. Tarel. Real time obstacle detection on non flat road geometry through 'v-disparity' representation. In *IEEE Intelligent Vehicles Symposium 2002*, pages 646–651, June 2002.

[LBCT03] J. Laneurit, C. Blanc, R. Chapuis, and L. Trassoudaine. Multisensorial data fusion for global vehicle and obstacles absolute positioning. In *IEEE Intelligent Vehicles Symposium*, 2003.

[LCCG07] B. Leibe, N. Cornelis, K. Cornelis, and L. Van Gool. Dynamic 3d scene analysis from a moving vehicle. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[LF97] Q. T. Luong and O. D. Faugeras. Camera calibration, scene motion and structure recovery from point correspondences and fundamental matrices. *International Journal of Computer Vision*, 22(3) :261–289, 1997.

[LHP80] H.C. Longuet-Higgins and K. Prazdny. The interpretation of a moving retinal image. In *Proceeding of Royal Society London*, volume B-208, pages 385–397, 1980.

[LK81] B.D. Lucas and T. Kanade. An iterative image registration technique with an application to stereovision. In *DARPA Image Understanding Workshop*, pages 121–130, April 1981.

[Low04] D. Lowe. Distinctive image features from scale-invariant key- points. *Int. Journal of Computer Visio*, 60(2) :91–110, 2004.

[LTD00] Z.N. Li, Z. Tauber, and M.S. Drew. Local-based object search under illumination change using chromaticity voting and elastic correlation. In *IEEE Conf. on Multimedia and Expo*, 2000.

[Lv00] B. P. L. Lo and S. A. velastin. Automatic congestion detection system for underground platforms. In *Proc. of Int. Symp. On Intell. Multimedia, Video and Speech processing*, pages 158–161, 2000.

[LZGR11] P. Lenz, J. Ziegler, A. Geiger, and M. Roser. Sparse scene flow segmentation for moving object detection in urban environments. In *IEEE Intelligent Vehicles Symposium*, 2011.

[Mac03] D. J.C. MacKay. *Information Theory, Inference, and Learning Algorithms.* 2003.

[Mai85] H. Maitre. Un panorama de la transformée de hough. *Traitement du signal*, 2(2) :305–317, 1985.

[MCUP02] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. In *British Machine Vision Conference*, pages 384–393, 2002.

[MG00] P. Monasse and F. Guichard. Fast computation of a contrast-invariant image representation. *IEEE Trans. on Image Processing*, 9(5) :860–872, 2000.

[MG06] S. Munder and D. M. Gravila. An experimental study on pedestrian classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28 :1863–1868, 2006.

[MJF94] W.J. MacLean, A.D. Jepson, and R.C. Frecker. Recovery of egomotion and segmentation of independant object motion using the em algorithme. In *British Machine Vision Conference*, pages 13–16, 1994.

[MN84] G. G. Medioni and R. Nevatia. Matching images using linear features. *IEEE Trans. on PAMI*, 6(6) :675–685, 1984.

[Mor81] H. P. Moravec. Rover visual obstacle avoidance. In *Proceedings of the 7th international joint conference on Artificial intelligence*, pages 785–790, august 1981.

[MS04] K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. *Int. Journal of Computer Vision*, 60(1) :63–86, 2004.

[MV98] J.B.A. Maintz and M.A. Viergever. A survey of medical image registration. *Medical Image Analysis*, 2(1) :1–36, 1998.

[Nac77] M. L. Nack. Rectification and registration of digital images and the effect of cloud detection. *Proc. Machine Processing of Remotely sensed data*, pages 12–23, 1977.

[NBT09] S. Nedevschi, S. Bota, and C. Tomiuc. Stereo-based pedestrian detection for collision-avoidance applications. *IEEE Intelligent Transportation Systems*, 10(3) :380–391, 2009.

[NH89] S. Negahdaripour and B.K.P Horn. A direct method for locating the focus of expansion. *Computer Vision Graphics and Image Processing*, 46(3) :303–326, 1989.

[OC73] F. O'Gorman and M. B. Clowes. Finding picture edges through collinearity of feature points. In *3rd Int. Joint Conf. Art. Int.*, pages 543–555, 1973.

[OPR78] R. Ohlander, K. Price, and D. R. Reddy. Picture segmentation using a recursive region splitting method. *Computer Graphics and Image Processing*, 8 :313–333, 1978.

[ORP00] N. M. Oliver, B. Rosario, and A. P. Pentland. A bayesian computer vision system for modeling human interactions. *IEEE Trans. on Patt. Anal. and Machine Intelligence*, 22(8) :831–843, 2000.

[PCSW89] C.A. Pelizzari, G.T.Y. Chen, D.R. Spelbring, and R.R. Weichselbaum. Accurate three dimensional registration of ct, pet and/or mr images of the brain. *Journal of computer assisted tomography*, 13(1) :20–26, 1989.

[PD98] N. Paragios and R. Deriche. A pde-based level set approach for detection and tracking of moving objects. In *Proc. of International Conference on Computer Vision*, pages 1139–1145, 1998.

[PJL+98] G.P. Penney, J.Weese, J.A. Little, P. Desmedt, D.L.G. Hill, and D. J. Hawkes. A comparison of similarity measures for use in 2d/3d medical image registration. *IEEE trans. Med. Image*, 17(4) :586–595, 1998.

[PK92] C. J. Poelman and T. Kanade. A paraperspective factorization method for shape and motion recovery. Technical report, Carnegie mellon university CMU-CS-91-208, 1992.

[PKF07] J. Pons, R. Keriven, and O. Faugeras. Multi-view stereo reconstruction and scene flow estimation with a global image-based matching score. *International Journal of Computer Vision*, 72(2) :179–193, 2007.

[RC96] B.S. Reddy and B.N. Chatterji. An fft-based technique for translation, rotation and scale-invariant image registration. *IEEE Trans. on Image Processing*, 5(8) :1266–1271, 1996.

[RFBC10] F. Rodriguez, V. Frémont, P. Bonnifait, and V. Cherfaoui. Visual confirmation of mobile objects tracked by a multi-layer lidar. In *IEEE International Conference on Intelligent Transportation Systems*, 2010.

[RK82] A. Rosenfeld and A. C. Kak. *Digital picture processing*. Academic Press, 1982.

[SB97] S. M. Smith and J. M. Brady. Susan : A new approach to low level image processing. *International Journal of Computer Vision*, 23(1) :45–78, 1997.

[SFS09]]. D. Scaramuzza, F. Fraundorfer, and R Siegwart. Real-time monocular visual odometry for on-road vehicles with 1-point ransac. In *International Conference on Robotics and Automation*, 2009.

[SG00a] C. Stauffer and W. E. L. Grimson. Learning patterns of activity using real-time tracking. *IEEE Trans. on Patt. Anal. and Machine Intelligence*, 22(8) :747–757, 2000.

[SG00b] C. Stauffer and W. E. L. Grimson. Learning patterns of activity using real-time tracking. *IEEE Trans. on Patt. Anal. and Machine Intelligence*, 22(8) :747–757, 2000.

[SH90] L.G. Shapiro and R.M. Haralick. Matching relational structures using discrete relaxation. In Teaneck World Scientic Pub. Co., editor, *Syntactic and Structural Pattern Analysis Theory and Applications*. World Scientific Pub. Co., Teaneck, NJ, 1990.

[Sha78] S. D. Shapiro. Transform method of curve detection for textured image data. *IEEE Trans. Comp.*, C-27(3) :254–255, 1978.

[SJBK08] J. Suhr, H. Jung, K. Bae, and J. Kim. Outlier rejection for cameras on intelligent vehicles. *Pattern Recognition Letters*, 29 :828–840, 2008.

[SKB82] G.C. Stockman, S. Kopstein, and S. Benett. Matching images to models for registration and object detection via clustering. *Pattern Recognition and Image Analysis*, 4(3) :229–241, 1982.

[SMS00] G. P. Stein, O. Mano, and A. Shashua. A robust method for computing vehicle egomotion. In *IEEE Intelligent Vehicles Symposium*, pages 362–368, October 2000.

[SPA07] N. Soquet, M. Perrollaz, and D. Aubert. Free space estimation for autonomous navigation. In *5th International conference on Computer Vision Systems*, 2007.

[SRR04] D. Sazbon, H. Rotstein, and E. Rivlin. Finding the focus of expansion and estimating range using optical flow images and a matched filter. *Machine Vision and Applications*, 15(4) :229–236, October 2004.

[Sze94] R. Szeliski. Image mosaicing for telereality applications. In *Proc. of IEEE Workshop on applications of computer vision*, 1994.

[TD83] P.R. Thrift and S.M. Dunn. Approximating point-set images by line segments using a variation of the hough transform. *Computer Graphics and Image Processing*, 21(3) :383–394, 1983.

[TP91] M. A. Turk and A. P. Pentland. Face recognition using eigenfaces. In *Computer Vision and Pattern Recognition*, 1991.

[Vap99] V. Vapnik. An overview of statistical learning theory. *Neural Networks*, janv., 1999.

[VBA08] T. Vu, J. Burlet, and O. Aycard. Grid-based localization and online mapping with moving objects detection and tracking : new results. In *Intelligent Vehicles Symposium*, 2008.

[VJS05] P. Viola, M. J. Jones, and D Snow. Detecting pedestrians using patterns of motion and appearance. *International Journal of Computer Vision*, 63(2) :153–161, 2005.

[VP89] A. Verri and T.A. Poggio. Motion field and optical flow : Qualitative properties. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 11(5) :490–498, May 1989.

[WADP97] C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland. Pfinder :real-time tracking of the human body. *IEEE Trans. on Patt. Anal. and Machine Intelligence*, 19(7) :780–785, 1997.

[WRV+08] A. Wedel, C. Rabe, T. Vaudrey, T. Brox, and U. Franke. Efficient dense scene flow from sparse or dense stereo data. In *European Conference on Computer Vision*, 2008.

[XD92] S. Xu and P. E. Danielson. Robust estimation of focus of expansion and depth from high confidence optical flow. In *IAPR Workshop on Machine Vision Applications*, 1992.

[ZSZ00] K. Zemirli, G. Seetharaman, and B. Zavidovique. Stable matching or selective junction points grouping. In *IEEE Joint Conference on Information Sciences*, 2000.

Troisième partie

Publications annexées

Int J Comput Vis
DOI 10.1007/s11263-011-0475-6

c-Velocity: A Flow-Cumulating Uncalibrated Approach for 3D Plane Detection

Samia Bouchafa · Bertrand Zavidovique

Received: 12 March 2010 / Accepted: 7 June 2011

Abstract This paper deals with plane detection from a monocular image sequence without camera calibration or *a priori* knowledge about the egomotion. Within a framework of driver assistance applications, it is assumed that the 3D scene is a set of 3D planes. In this paper, the vision process considers obstacles, roads and buildings as planar structures. These planes are detected by exploiting iso-velocity curves after optical flow estimation. A Hough Transform-like frame called *c-velocity* was designed. This paper explains how this *c-velocity*, defined by analogy to the *v-disparity* in stereovision, can represent planes, regardless of their orientation and how this representation facilitates plane extraction. Under a translational camera motion, planar surfaces are transformed into specific parabolas of the *c-velocity* space. The error and robustness analysis of the proposed technique confirms that this cumulative approach is very efficient for making the detection more robust and coping with optical flow imprecision. Moreover, the results suggest that the concept could be generalized to the detection of other parameterized surfaces than planes.

Keywords Image motion analysis · Pattern recognition · Image scene analysis · Egomotion · Optical flow

1 Introduction

This work deals with 3D scene reconstruction from an onboard moving camera in the context of automatic driver assistance systems. Besides basic navigation and localization aspects, most proposed algorithms focus on obstacle detection. The obstacle is assumed to be a frontal plane, and real-time implemented approaches are generally based on stereo vision, especially when the acquisition system is well calibrated. Motion information is only exploited afterwards for detected objects. To take motion into account as information straight from image sequences, the egomotion of the camera is exploited to distinguish between various (moving) objects. Recent years have seen a profusion of work on 3D motion, egomotion or structure from motion estimation using a moving camera. One classification that is commonly accepted groups existing techniques into three main categories: discrete, continuous and direct approaches.

- Discrete approaches are based on matching and tracking primitives (point, contour lines, corners, etc.) extracted from images in sequence (Hartley 1995; Luong and Faugeras 1997; Bay et al. 2008). They are usually very effective. However, they suffer from a lack of truly reliable and stable features, e.g., time and viewpoint invariance. Moreover, in applications where the camera is mounted on a moving vehicle, homogeneous zones or linear marking on the ground hamper the extraction of reliable primitives.
- Continuous approaches exploit optical flow (Hildreth 1992; MacLean et al. 1994; Roberts et al. 2009). The relationship between the computed optical flow and real theoretical 3D motion allows, through optimization techniques, to estimate the motion parameters and depth at each point. Results are dependent on the quality of the computed optical flow.
- In direct approaches (Irani et al. 1997; Stein et al. 2000), motion is determined directly from the brightness invariance constraint without having to calculate explicitly an optical flow. Motion parameters are then deduced by conventional optimization approaches.

S. Bouchafa (✉) · B. Zavidovique
Institut d'Electronique Fondamentale, University Paris Sud XI,
91405 Orsay Cedex, France
e-mail: samia.bouchafa@u-psud.fr

Published online: 01 July 2011

Independent of the classification above, a large group of approaches—indifferently discrete, continuous or direct—exploit the parallax generated by motion (motion parallax, affine motion parallax, plane + parallax). These methods are based on the fact that depth discontinuities make it possible to separate camera rotation from translation (Irani et al. 1997; Hanes et al. 2008). For instance, in "plane + parallax" approaches, knowing the 2D motion of an image region where variations in depth are not significant can eliminate the camera rotation. Using the obtained residual motion parallax, translation can be exhibited easily.

Two particular studies drew our attention. In the first study (Fermuller and Aloimonos 1995), motion vectors of certain lengths and directions are constrained to lie on the image at particular loci whose location and form depend solely on the 3D motion parameters. If optical flow fields or stereo disparity fields are considered, then equal vectors are shown to lie on conic sections. By studying various properties of these curves and regions and their relationships, a characterization of the structure of rigid motion fields could be made. In the second study (Labayrade et al. 2002), the authors propose a very efficient stereo vision technique based on the *v-disparity* concept that consists of dealing with the relation between disparity and image lines, in the particular case where images are rectified. The new projection space—*v-disparity* space—where the relation appears, builds on the set of line disparity histograms. These two studies can be paralleled in a very interesting way: both exploit iso-value curves—velocity or disparity. Our opinion is that such a process, mixing iso-value curves and statistics, is generalizable. This paper shows how to transpose the concepts to motion and how practically to implement this theory.

For a practical illustration, parametrized surfaces were detected without camera calibration or *a priori* knowledge about the vehicle egomotion. To circumvent the problem of depth estimation, in this study, it is assumed that the 3D scene is a set of 3D planes. They will thus be detected by exploiting iso-velocity curves that make velocity structures emerge out of an optical flow estimation. Experiments have already been performed, and it is commonly agreed in physics that cumulative approaches (here, voting scheme) are very efficient (Bouchafa and Zavidovique 2006) in making a detection more robust and dealing with major (here, optical flow) imprecision. Therefore, it is shown how to extend the *v-disparity* technique to detect planes along an image sequence shot from a moving vehicle. The apparent velocity from the scale change occurring to image data takes the place of the disparity, leading to the so-called *c-velocity* frame (Bouchafa and Zavidovique 2008). In this paper, a complete plane detection process is detailed, and the various sources of uncertainty are evaluated.

The paper is organized as follows: the next section is devoted to the computation of constant velocity curves in the image plane and the cumulative process is explained. The third section explains how the focus of expansion is found from the optical flow. The fourth section details the parabolas—3D planes—extraction in the *c-velocity* space using a Hough transform enriched by a k-means technique. Section 5 is devoted to the results. Then, in Sect. 6, all constraints are specified, and the issues relating to our proposed method are analyzed.

2 A New Concept: *c-Velocity*

In Labayrade et al. (2002), the authors exploit the intuition and prove that, along a line of a stereo pair of rectified images, the disparity is constant and varies linearly over a horizontal plane as a function of the depth. Then, by considering the mode of the 2D histogram of disparity value vs. line index, i.e., the so called *v-disparity* frame, the features of the straight line of modes indicates the road plane, for instance. The computation is then generalized to the other image coordinate and to vertical planes in using the *u-disparity*, by several teams, including ours on our autonomous car. In this paper, the latter concept is transposed to motion. These computations build on the fact that any change in position of a camera results in an apparent shift of pixels between images: that is, a disparity for a stereo pair and velocity for an image sequence. The *v-disparity* space draws its justification, after image rectification that preserves horizontal—iso-disparity—lines, from inverse-proportional relations between the first image horizontal-line positions vs. depth and second depth vs. disparity. How to derive the same type of relation in the egomotion case is shown here.

2.1 The Case of a Moving Point

In this paper, a translational rigid straight move of the camera in the Z direction is assumed. It does not restrict the generality of computations. Then, as a result of using Appendix A, (14) with $\Omega_X = \Omega_Y = \Omega_Z = T_X = T_Y = 0$, the 2D velocity (u, v) becomes:

$$\begin{cases} u = \frac{T_Z}{Z} x \\ v = \frac{T_Z}{Z} y \end{cases} \tag{1}$$

Equations (1) describe a 2D motion field that could be approximated by the optical flow. To tackle the imprecision of optical flow velocity vectors, a Hough Transform-like projection space is defined, which—thanks to its cumulative nature—can perform a robust plane detection.

From (1), the relation between the velocity $\|\mathbf{w}\|$ ($\approxeq$ disparity) and the iso-velocity function index c ($\approxeq$ line index v) becomes

$$\|\mathbf{w}\| = \sqrt{u^2 + v^2} = \left| \frac{T_Z}{Z} \right| \sqrt{x^2 + y^2} = K \cdot C(x, y) \tag{2}$$

$$\frac{\|\mathbf{w}\|}{K} = C(x, y) = c \tag{3}$$

The translation T_Z is that of the camera, and it is identical for all static points. Then, if the depth Z is constant, K, defined as $|\frac{T_Z}{Z}|$ in (2) is constant, and the iso-velocity curves $C(x, y)$ are circles. Moreover, the radius c varies linearly with the velocity norm $\|\mathbf{w}\|$ as underlined by (3). Beyond the "punctual" general case, Z can be eliminated by considering linear relations with (X, Y), i.e., plane surfaces fitting the driving application well.

2.2 The Case of a Moving Plane

Four cases of moving planes were studied:

(a) Horizontal (road)
(b) Lateral (buildings)
(c) Frontal1 (fleeing/approaching obstacle)
(d) Frontal2 (crossing obstacle)

Table 1 lists, for each case, the unit normal vector $\mathbf{n}$, the 3D translation vector $\mathbf{T}$ and the distance plane-to-origin d. The camera is assumed to have a translational motion $\mathbf{T} = (0, 0, T_Z)$. In this case, in (15) Appendix B, T_X, T_Y, Ω_X, Ω_Y and Ω_Z are set to zero (except for an obstacle with its own motion $(0, 0, T_Z^o)$ or $(T_X^o, 0, 0)$ that adds to $\mathbf{T}$ giving $\mathbf{T}'$).

The corresponding motion fields are then obtained by injecting the respective $\mathbf{T}$ (or $\mathbf{T}'$) and $\mathbf{n}$ into a_i for $i = 1, \ldots, 8$. Eventually, (15) outputs u and v, as listed in Table 2 for each case. Let $\|\mathbf{w}_o\|$, $\|\mathbf{w}_r\|$, and $\|\mathbf{w}_b\|$ be respectively the module of the apparent velocity of an obstacle point, a road point and a building point. All extrinsic and intrinsic parameters are grouped into an unknown constant factor K. The only interest of K at that stage is to be constant to reveal a plane.

Table 1 Plane parameters of four relevant cases of moving planes

	$\mathbf{n}$	3D motion	Dist. to origin
(a)	$(0, 1, 0)$	$\mathbf{T} = (0, 0, T_Z)$	d_r
(b)	$(1, 0, 0)$	$\mathbf{T} = (0, 0, T_Z)$	d_b
(c)	$(0, 0, 1)$	$\mathbf{T}' = (0, 0, T_Z^o + T_Z)$	d_o
(d)	$(0, 0, 1)$	$\mathbf{T}' = (T_X^o, 0, T_Z)$	d_o

Table 2 Motion velocity vectors of four relevant cases of moving planes

(a)	$u = \frac{T_Z}{f d_r} xy$ $v = \frac{T_Z}{f d_r} y^2$	$\|\mathbf{w}_r\| = K\sqrt{y^4 + x^2 y^2}$
(b)	$u = \frac{T_Z}{f d_b} x^2$ $v = \frac{T_Z}{f d_b} xy$	$\|\mathbf{w}_b\| = K\sqrt{x^4 + x^2 y^2}$
(c)	$u = \frac{T_Z + T_Z^o}{d_o} x$ $v = \frac{T_Z + T_Z^o}{f d_o} y$	$\|\mathbf{w}_o\| = K\sqrt{x^2 + y^2}$
(d)	$u = \frac{T_Z}{d_o} x - \frac{T_X^o f}{d_o}$ $v = \frac{T_Z}{d_o} y$	$\|\mathbf{w}_o\| = \begin{cases} K \text{ if } T_X^o \ggg T_Z \\ K\sqrt{x^2 + y^2} \text{else} \end{cases}$

2.3 Exhibiting the Proportionality $c/\|\mathbf{w}\|$

Each type of $\|\mathbf{w}\|$ leads to the corresponding expression of c and the related iso-velocity curve, from (a) to (d) in Table 2. For instance, in case (b) of a building plane,

$$c = \frac{\|\mathbf{w}\|}{K} = \sqrt{x^4 + x^2 y^2} \tag{4}$$

Figure 1a displays the iso-velocity curve for a given value c_0 of c, in the horizontal plane case: case (a) of Table 2. It is the set of pixels where the velocity norm is constant, $\mathbf{w} = K c_0$, if they belong to the image of a horizontal plane. Indeed, the formula (4) above proves that c, constant along iso-velocity curves by definition, is proportional to $\|\mathbf{w}\|$. Actually, an horizontal plane in the image intersects the family of such curves obtained when c varies, as displayed in Fig. 1b incrementing here c values by 10.

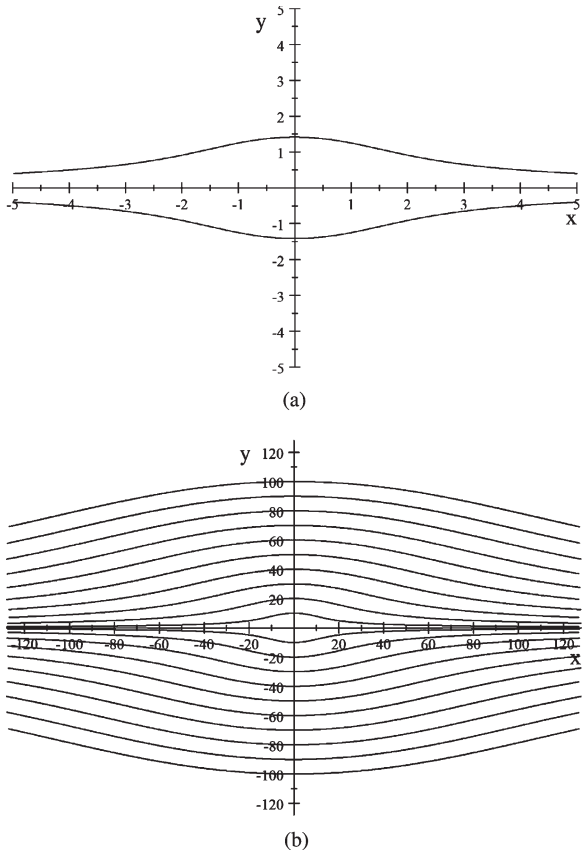


Fig. 1 (**a**) A couple of curves for a given c value in the case of a road model, on both sides of the origin. The accumulation is done along these curves, which theoretically do not intersect. (**b**) A set of curves for step 10

2.4 Displaying Proportionality Towards Plane Extraction

The model above, as built in Sects. 2.2 and 2.3, shows that 2D velocities, should be constant along c curves corresponding to the adequate plane model, whatever the process to find them. Given the uncertainty and imprecision in image acquisition and 2D motion estimation, $\|\mathbf{w}\|$ is likely not constant and the most represented $\|\mathbf{w}\|$ value along a given c curve is chosen as the winner. In collecting all winners for every c value, proportionality should pop out within a Hough Transform-like process. Thus, as explained in the introduction, the *c-velocity* space will be a cumulative space framed in coordinates $(c, \|\mathbf{w}\|)$. It is constructed by assigning to each pixel (x, y) the corresponding c value through the chosen plane model—(a), (b), (c) or (d) in Table 2—and in incrementing the $(c, \|\mathbf{w}\|)$ cell value, where $\mathbf{w}$ is the velocity found in (x, y). In the current experiments, the latter $\mathbf{w}$ was computed with a classical optical flow method (Lucas and Kanade 1981).

2.5 Numerical Technicalities

A study of the function $c(x, y)$ that corresponds to each plane model—in particular for the road and the building model—leads to the following conclusions: first, each previous curve intersects the x axis (road model) or y axis (building model) in the image plane within: $x = \pm\sqrt{c}$ or $y = \pm\sqrt{c}$, respectively. Second, for a standard image size, the range of variation of c is very large, actually 128000 (road model) and 96000 (building model) for an image size 320×240, respectively. As a consequence, for implementation (computing accuracy) and homogeneity ($c \approx \text{length}^2$), these two models are translated into the relations between $\|\mathbf{w}\|$ and $\sqrt{c}$. A plane is then represented in the *c-velocity* space by a parabola instead of a line.

Remark The choice is not insignificant considering $\sqrt{c}$ in the discrete space of the image where pixel matter amounts to merge all $c + \delta$ curves such that $l \leq \sqrt{c + \delta} \leq l + 1$. Because $\|\mathbf{w}\|$ is roughly sampled both in accordance with the method's philosophy and due to the optical flow smoothing process, it limits the precision on velocities (see Sect. 6.1 and Fig. 11).

2.6 Cumulative Curves Rectification

For each point $\mathbf{p} = (x, y)$ in the image, there is an associated c value depending on the chosen plane model (see right column of Fig. 2). It is computed once and off-line because it depends only on (x, y). In addition, it is possible for implementation facilities and by analogy to image rectification (that makes all epipolar lines parallel) to compute the transformation that makes all the c-curves parallel to the image line, that is the intensity function $I(c, y)$ for the road and obstacle models and $I(x, c)$ for the building model (see Appendix C).

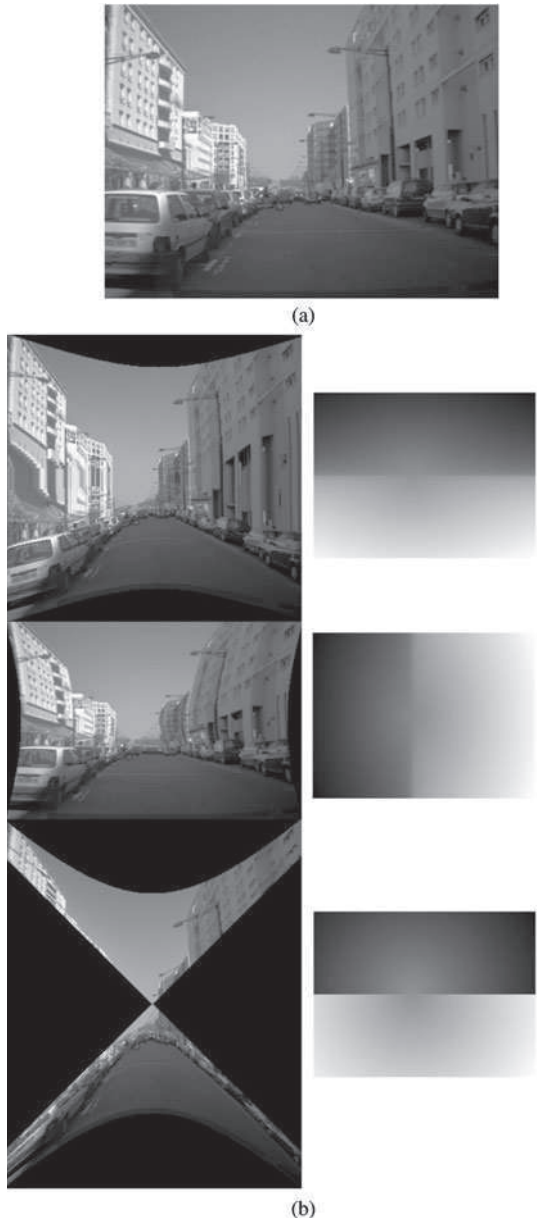


Fig. 2 (**a**) Original image. (**b**) *Left*: images that are constructed using the geometric transformation that makes all c-curves parallel straight lines. *Right*: the c-map, each image point gets the c-value for, respectively, the road, building and obstacle models

3 FOE Extraction

Out of convention, the intersection of the focal plane and the motion direction, i.e., the Focus Of Expansion, is the origin of the 2D image frame. Its location is thus a key parameter of the visual process studied here. In the case of a transla-

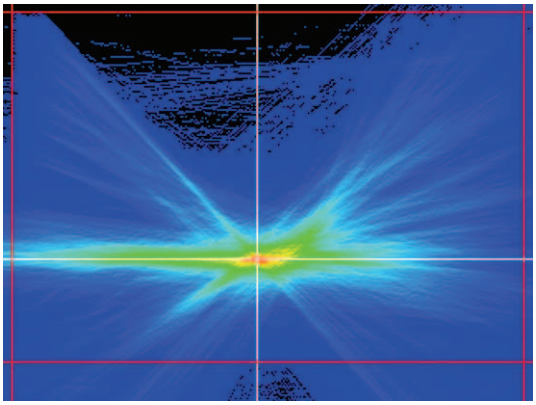


Fig. 3 (Color online) Voting space for FOE determination. In *red*: point that cumulates maximum votes. The intersection of the *two white lines* is the FOE. For each pixel location, its voting rate is associated with a color from *Dark* (no vote) to *red* (maximum votes). Intermediate colors are (in order) *blue*, *cyan* and *yellow*

tional motion, each velocity vector points toward the FOE as, from (1),

$$\frac{u}{v} = \frac{x}{y} \tag{5}$$

Assume now that (x_0, y_0) are the FOE coordinates in the image. Moreover, the origin of the image coordinates system is placed on the top left corner of the image, and then the egomotion (u, v) becomes

$$\begin{cases} u = \frac{T_Z}{Z}(y_0 - y) \\ v = \frac{T_Z}{Z}(x - x_0) \end{cases} \tag{6}$$

and

$$\theta = \tan^{-1}\left(\frac{v}{u}\right) = \tan^{-1}\left(\frac{x - x_0}{y_0 - y}\right) \tag{7}$$

The above relation means that one can extract the FOE by estimating the intersection of all velocity vector lines. Several other methods exist (Sazbon et al. 2004; Negahdaripour and Horn 1989). For the sake of further real on board implementation, a method that is coherent with the present computations is favored (Bouchafa and Zavidovique 2006). All pixels are asked to vote for a global intersection point of apparent velocity vectors within a regular Hough space. In practice, each velocity vector votes for all the points belonging to its support line. The FOE corresponds then to the point with maximum votes (see Fig. 3).

The location of the FOE can confirm or not whether the main hypothesis is valid (translational motion). When the FOE is not located at the image center, images need to be rectified to compensate for a possible pan or tilt of the camera (see Sect. 6.3). Note also that (5) could also serve as additional constraint on voters of the *c-velocity* frame.

4 Extracting Parabolas Using a 1D Hough Transform

Planes are represented in the *c-velocity* space by parabolas. They can be extracted with a Hough transform again. The distance p between each parabola and its focus or its directrix is then cumulated in a one-dimensional Hough transform (Duda and PE 1972).

$$(x, y, \|\mathbf{w}\|) \rightarrow (c, \|\mathbf{w}\|, P(c, \|\mathbf{w}\|)) \rightarrow P(p)$$

where P is the probability, and p is the parabola's parameter. The classes of the histogram split through a k-means clustering (MacKay 2003). Of course, other clustering approaches would suffice.

$$\|\mathbf{w}\| = K(\sqrt{c})^2 \quad \Rightarrow \quad p = \frac{1}{4K} = \frac{(\sqrt{c})^2}{4\|\mathbf{w}\|} \tag{8}$$

Each 3D plane corresponds to a parabola (a given p parameter). Of course, different kinds of perturbations like errors on plane orientations or on the camera pose are inherent to the process. The hypothesis of a translational motion or the estimations of the velocities (FOE shift) may be wrong. Finally, the inter-model perturbation cannot be avoided (see Sect. 6.5). All perturbations concur to transform a pick in the histogram into a wider distribution. This is the main reason why the 1D histogram needs clustering. With k-means, the number of clusters is set *a priori*. It is not a real difficulty in targeted applications because that number is usually limited through the scene structure and the due precision. For instance, urban motion leads to the detection of only one "road" but 4 "buildings" (left/right true buildings + parked car sides). Obviously, such a prediction may turn out to be wrong. In that case, three considerations hold:

- Assume the number of planes is actually greater. Some clusters close in orientation will merge, and the question becomes whether it lowers the car driving accuracy (e.g., would a car run out of the road based on an averaged distance to the walls?). This question is among the reasons for studying perturbations (see Sect. 6).
- Whether the preset number of planes is lower or greater, can it be corrected? With elementary supervised clustering as k-means, one can always loop on this number and control from the very likelihood of results. For instance, in the image of Fig. 5, predicting one class more will separate the street light from the right car sides. For a more quantitative flavor, one can refer to Fig. 7 where a marked line on the road first appears and a second can be further clustered from the second obstacle (motorcycle).
- This problem is a classical dilemma to trade off between unsupervised likely complicated clustering and a more elementary one but with *a priori* knowledge. In this preliminary study, to remain more focused on the principle of this method, the second choice was favored.

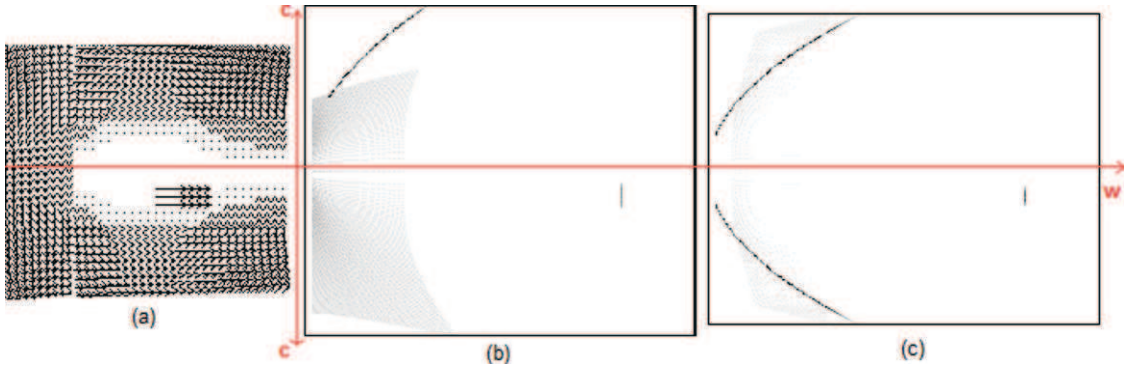


Fig. 4 (**a**) Velocity vector field of a moving scene with a building, a road and an obstacle plane. Vectors figure the optical flow. For clearer display, they are sub-sampled in the image space and their amplitude is trimmed. Note that the same display process is used for all optical flow images in the paper, whether synthetic or real. (**b**) and (**c**) Associated *c-velocity* spaces (*left*: building, *right*: road)

Remark "Frontal obstacle planes" result into a vertical segment of the $(\sqrt{c}, \|\mathbf{w}\|)$ cumulative space, regardless of the model being considered (see Fig. 4), because the velocity is constant. It is still true from experiments because of the sub-sampling of $\|\mathbf{w}\|$ (optical flow) and the likely small size of obstacles when they are fleeing or the egomotion can be neglected.

5 Results of Plane Detection

5.1 Synthetic Images

In the following toy example (Fig. 4a), a synthetic velocity vectors field of a moving 3D scene is generated with 3 planes: a vertical one (on the left of the image), a horizontal one (on the bottom of the image) and a frontal plane with its own motion parameters (a crossing obstacle).

The results confirm that this simulated egomotion transforms a road plane and a building plane into a parabola in the corresponding *c-velocity* space. In Fig. 4b, the partial parabola indicates the expected moving plane together with its limited width and asymmetrical position. Indeed, the extension of the piece of the parabola is proportional to the latter width. Note the *moiré* effect around the origin in the building: the inter-model signature is predominant there, also due to the respective voting amplitudes. In Fig. 4c, the full parabola indicates the expected moving plane. The gap near the origin corresponds to the weak velocity amplitude near the FOE still amplified by the discretization. Note that the *moiré* effect is still there but less pronounced than in Fig. 4b due to the lower number of voting pixels from the building model. Eventually, the constant segment figures the obstacle, which appears in all *c-velocity* spaces because of its constant velocity $\mathbf{w}$ (cf. Remark above).

5.2 Real Images

5.2.1 Data and Parameters

All image sequences considered for the experiments stem from the French project LOVe (Logiciel d'Observation des VulnérablEs). Various sequences of real car driving in urban scenes, provided by the car making companies involved in the project, with different kinds of vehicle motions are stored in this database. The process needs an optical flow estimation, and the classical Lucas and Kanade (1981) method was chosen with a 9×9 window size. The latter choice is not critical because the aim of this study is a robust detection despite velocity field uncertainties. Result examples put a major stress on the building *c-velocity* space in this paper in relation to the urban nature of the sequences.

Two more parameters are involved. The first one refers to the normalization with respect to the total number of points in a c-curve: indeed, due to the digitization, curves have various numbers of elements (see Sect. 6 and concluding remarks in Sect. 2.5). The second one is the minimum voting rate in the 1D Hough transform cell for parabola extraction: for a conic, one must trivially have at least three *c-velocity* cells to assume the possible existence of a curve passing through. Then, for a quantitative evaluation of the approach, two kinds of confidence factors are used. The first one is related to the translational motion hypothesis: it is the difference Δ_{foe} between the coordinates of the image center and the position of the focus of expansion. If Δ_{foe} is large, it means that the vehicle does not follow a straight trajectory. In this situation, it should be difficult to detect any parabola in the *c-velocity* space. The second one is related to possible contamination. The standard deviation σ of each k-means cluster is chosen. Points far from the mean may for instance belong to another plane model. A small σ together with a big amplitude in the *c-velocity* space is a guarantee that a plane really exists in the image. Indeed, the total number of voters is the surface of the plane.

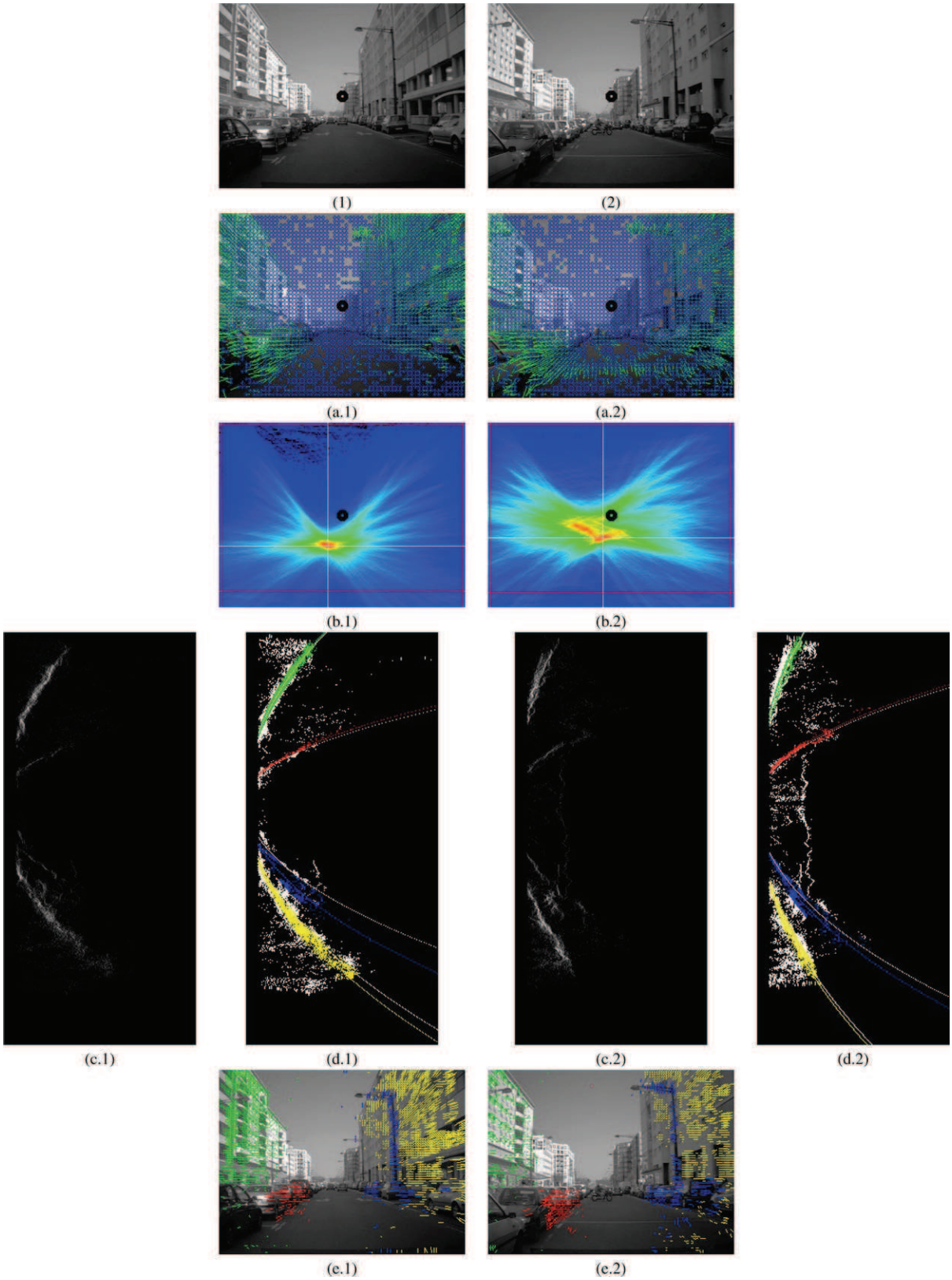


Fig. 5 Some typical results. Image (**1**) corresponds to the case where the camera moves straight. Image (**2**) corresponds to the case where a motorcycle crosses the road. In image (**3**), the vehicle stops. For each case, images (**b**) give the result of the FOE determination, images (**c**) the *c-velocity* building space, images (**d**): the *c-velocity* space after *k*-means clustering and images (**e**) the final results with 3D lateral plane detection

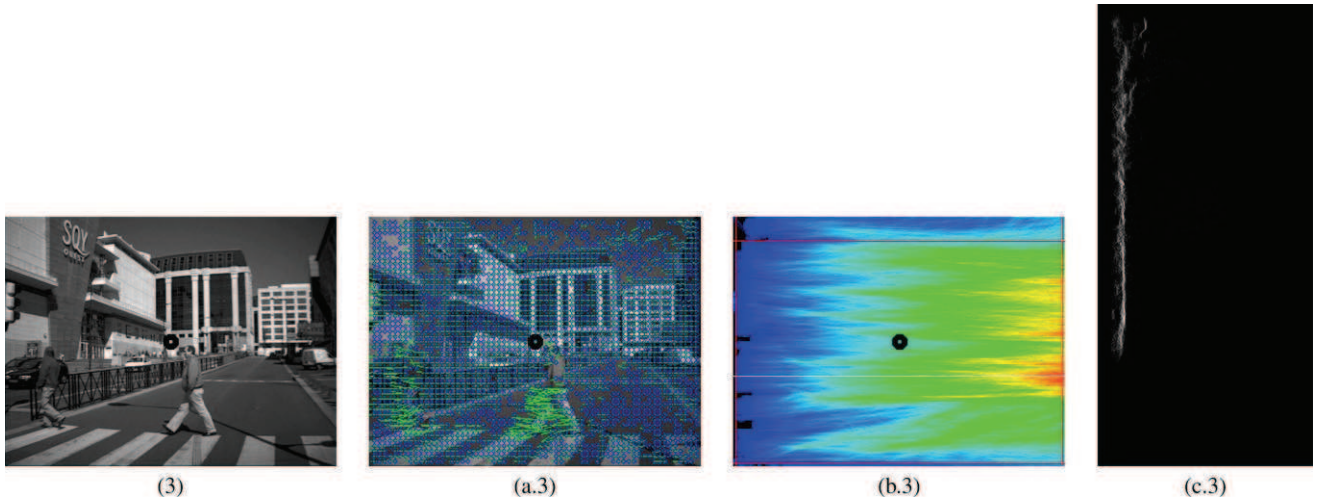


Fig. 6 Image (**3**): a case where the vehicle stops. (**3.a**) gives the optical flow, (**3.b**) the result of the FOE determination, and (**3.c**) the *c-velocity* building space

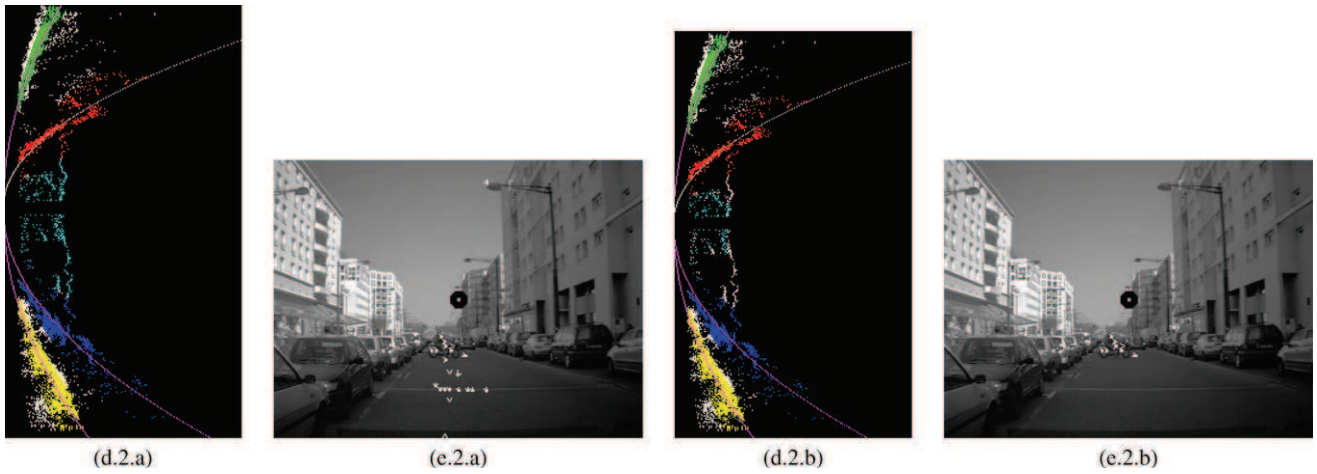


Fig. 7 Results below show how to make use of the *c-velocity* concept to detect obstacles

5.2.2 Results

The examples of Fig. 5, 6, 8 and 9 display the optical flow (images labeled "a") subsampled for sake of readability, the FOE position (images b) and the building *c-velocity* spaces from various image sequences (images c). Results of the parabola extraction are given (images d) with the corresponding planes in the image (images e) if they exist. Some particular situations are shown as well, where the camera rotates or stalls. The examples show how the approach could be used for obstacle detection even though it is not the main goal of this study.

5.2.3 Interpretation

In all the sequences considered, the circle in black is the center of the image. Δ_{foe} is its distance from the FOE. In images 1 and 2, one can see 6 moving planes: 2 planes corresponding to buildings, 2 planes corresponding to cars parked on the sides, a frontal moving obstacle (a motorcycle crossing the road) and the road plane. In this sequence, velocity vectors are in the majority on vertical planes. In the building *c-velocity* space in (d.1) and (d.2), as expected, four parabolas corresponding to the four main vertical planes in the sequence are obtained. Planes in (e.1) and (e.2) have a label according to a 4 class k-means clustering. The same colors are used for parabola display and the corresponding plane display in the image. The discarded points are displayed in white they most likely belong to another plane model. Image 3 gives an example in the case where the camera does not move. Pedestrians are crossing the road, and the corresponding *c-velocity* space (c.3) gives a line (constant velocity).

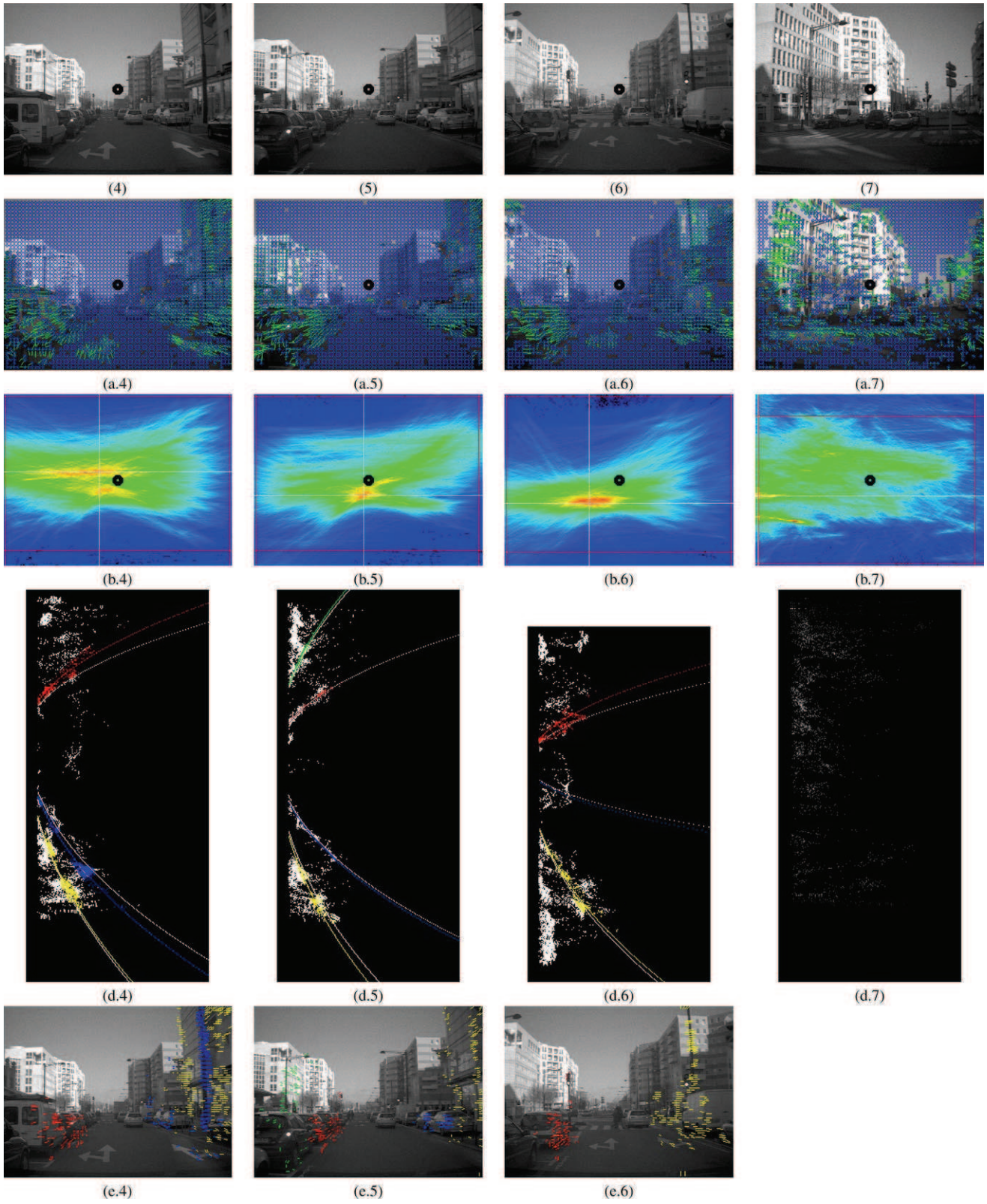


Fig. 8 More results in some cases where the number of planes varies (images **4**, **5**, **6**) and the camera rotates (image **7**)

In the example of Fig. 7, it is shown how image (2) below could be used for obstacle detection. If discarded points in the corresponding *c-velocity* space (points in cyan in d.2.a) are selected, considering only those points between the two parabolas on both sides of the FOE and which points voted for them in the image, then vectors in white in the image (e.2.a) are obtained. Of course, all connected points that have constant velocity are highlighted (the mark on the road gives a constant velocity and is detected also with the crossing motorcycle). If only points that have a given direction of motion (crossing) in (d.2.b) are selected—points in cyan—then only the motorcycle is extracted from image (e.2.b).

Using images 4, 5 and 6 (see Fig. 8), as expected, respectively 3, 4 and 2 parabolas are obtained in the *c-velocity* spaces d.4, d.5 and d.6. They correspond respectively to 3, 4 and 2 vertical planes in the images (see, e.4, e.5 and e.6) according to the information from the optical flow. In image 7, the camera rotates around the Y axis, and no parabola appears in the *c-velocity* space d.7.

In the sequence of Fig. 9, a truck is crossing the road in image 8. Thus, only the building on the right is detected (see e.8), corresponding to only one parabola in (d.8). In image 9, instead of a building, a car is detected on the right that runs with the same motion as the camera-equipped vehicle. It is then considered as a vertical plane with the same relative motion (see e.9).

For each image displayed above, Table 3 gives some relevant information to show the quality of the results for Δ_{foe}, σ_i where i is the parabola number and

$$\kappa_i = 1 - \frac{\text{number of connected components of the region}}{\text{number of pixels within the region}},$$

which corresponds to a connexity factor. Indeed, each detected parabola is associated with a connectivity value, giving connectivity information about the corresponding pixels in the image. A plane in the image is assumed to be a connected compound of pixels. Then, a parabola with a connectivity number close to 1 translates a plane region in the image that is strongly connected.

Let us consider now the sequence of 2300 images, from which images 1 and 2 were extracted. The 500 images that correspond to the phase where the camera moves globally straight and the scene is composed of 4 main planes are extracted. Figure 10 plots, for each image, the couple $(\Delta_{foe}, \overline{\sigma_i})$. The higher Δ_{foe} , the higher $\overline{\sigma_i}$, which confirms the role of the FOE position. Note that the apparent translation and the FOE position are linked by the observation: a wrong translational motion causes the parabola to spread into a cluster. Along this sequence, the parabola detection rate is equal to **87%**: the 4 planes are detected. The remaining 13% are the cases where at least one plane is missing.

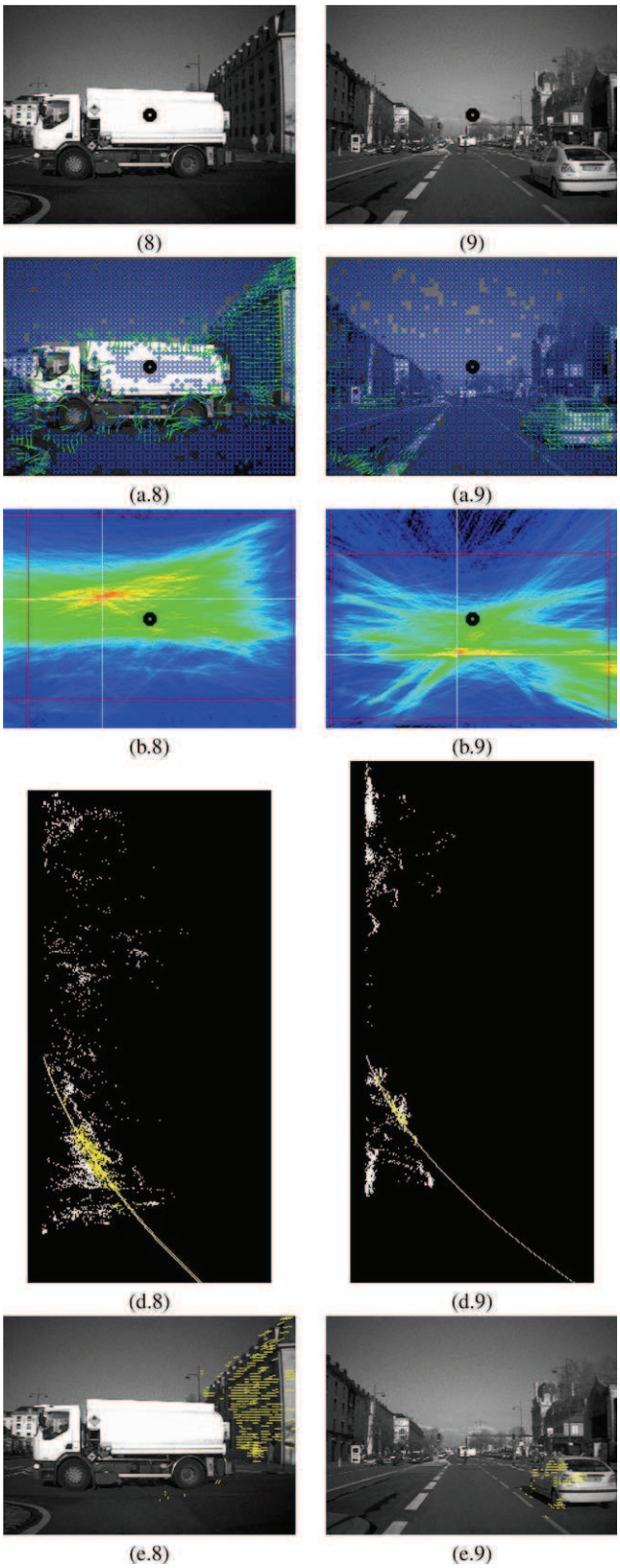


Fig. 9 Peculiar cases: a single obstacle (image **8**) and a vehicle with the same relative motion (image **9**)

Table 3 Indicators for appreciation of the results quality

	Δ_{foe}	σ_1	σ_2	σ_3	σ_4	κ_1	κ_2	κ_3	κ_4
1	31	2	11	11	12	0.76	0.98	0.82	0.96
2	44	4	25	13	20	0.83	0.93	0.85	0.97
3	162	–	–	–	–	–	–	–	–
4	28	8	–	14	22	0.75	–	0.81	0.91
5	22	2	11	2	26	0.92	0.76	0.9	0.74
6	52	12	–	–	15	0.81	–	–	0.84
7	158	–	–	–	–	–	–	–	–
8	56	–	–	–	22	–	–	–	0.87
9	42	–	–	–	10	–	–	–	0.78

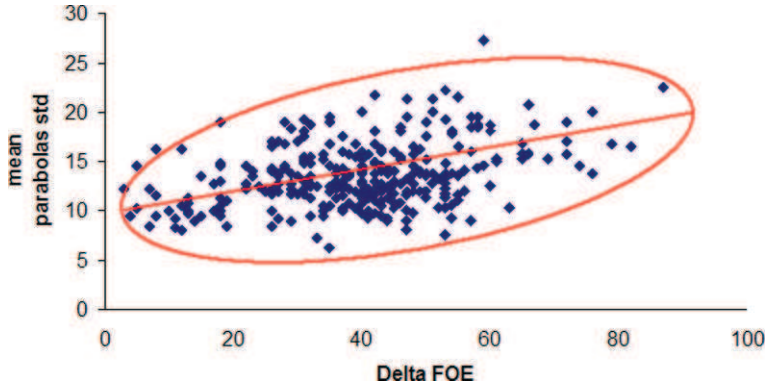


Fig. 10 For each image of a sequence, the couple $(\Delta_{foe}, \overline{\sigma_i}_{,i=1,2,3,4})$ is plotted

6 Robustness analysis

Let us consider 5 sources of ambiguity or imprecision in the plane finding process described in the paper:

1. Numerical approximations.
2. Noisy optical flow.
3. Disalignment of the camera wrt. the translation (e.g. camera with pan to ground).
4. Wrong estimation of the FOE.
5. Inter model contamination.

Sources 1 and 2 are less bound to the extraction technique studied here. Digital image analysis involves integer calculus, and mobile detection involves velocity. They are treated first and independently. For sources 3 to 5, before completing any error calculus, it must be stressed upon the cascade of 2 spaces, the second figuring a measure (i.e., probability) on the first, where all errors arise.

- The four error sources work in the (x, y) image space, causing a pixel considered not to be at the position or not to have the exact size it is expected to have. Ultimately, the error ends affects the theoretical iso-velocity lines $c(x, y)$.
- However, the plane estimation is actually performed in the $(c, \|\mathbf{w}\|)$ space, where proportionality is looked for through some consensus of couples $(c, \|\mathbf{w}\|)$. The consensus is measured as the proportion of pixels along the c line that show a $\|\mathbf{w}\|$ velocity.

In other words, if some perturbation of the theoretical iso-velocity line c occurs, then constant $\|\mathbf{w}\|$ are sought on the theoretical c-curves while $\|\mathbf{w}\|$ is actually constant on the perturbed curves. Depending on the type of perturbation, three main consequences need further attention.

Result 1 Any error from source 3 or 4 results in some shift of the theoretical iso-velocity line c. Then, sub populations that would support the proportionality will actually be at the intersection of the theoretical and actual lines, as illustrated Fig. 14. As a consequence, not do histogram amplitudes drop but also the contribution of a given $\|\mathbf{w}\|$ depends on the c line density, to know how many potential voters could be in its favor.

Result 2 In cases 3 and 4, the shift involves two limit curves in the $(\sqrt{c}, \|\mathbf{w}\|)$ frame for intersections to exist in the (x, y) frame (Fig. 14). It can be proven that these limit curves maintain a parabolic form. Additionally, one should not forget the actual limits of the planar objects under consideration and the image bounds.

Result 3 Whatever the error type 3, 4 or 5, from $c = x\rho$ or $y\rho$ it is obvious that δc is decreasing relatively with the

distance—$\rho(=\sqrt{x^2+y^2})$—to the FOE, and then any error causes more perturbation close to it. For larger ρ values, the curve length likely increases, thus potentially decreasing the relative importance of a given $\|\mathbf{w}\|$ because of the scene structure (compact planes). In same conditions, because curves flatten asymptotically, the increased intersection increases the set of voters in favor of the related $\|\mathbf{w}\|$, right or wrong. A converging consequence is that the fuzz or *moiré* effects due to errors occurs mostly close to the FOE, as shown in the simple cases in Fig. 4 and Fig. 16.

6.1 Numerical Approximations

The equation of *c-velocity* curves involve a fourth polynomial degree of which consequences were already drawn in the closing Remark of 4. The more precise effect is illustrated by Fig. 11 where contiguous curves $\sqrt{c} = \sqrt[4]{y^4+x^2y^2}$ are displayed in alternating black, gray and white for the sake of understanding. Three phenomena impact the voting population:

- The increasing curve length function of y (small around the FOE and then truncated to the image dimensions).
- The pseudo-periodical thickness variation with y.
- The pseudo-periodical dotted line effect (fortunately vanishing with y increasing due to the bounded image dimension).

Fig. 11 Contiguous ($\sqrt{c} \rightarrow \sqrt{c}+1$) curves $\sqrt{c} = \sqrt[4]{y^4+x^2y^2}$ are displayed in alternating *black*, *gray* and *white* to illustrate their parallelism in the image. This figure is a discrete real version of continuous Fig. 1

Moreover, the effects depend on the real to integer conversion process and could be optimized if useful.

6.2 Optical Flow Imprecision

All other functions involved in the process (parallax equations, FOE location, *c-velocity* curves, etc.) can be considered to be deterministic. Then, a stochastic additive noise would transfer straight onto these functions. For instance, let us introduce a Gaussian noise on the optical flow vectors of a building. The parabolas in the voting spaces are now thicker, and their thicknesses depends trivially on the noise variance (see Fig. 12; the law of probability of $\sqrt{c}$ is the law of $\sqrt{\|\mathbf{w}\|}$).

6.3 Error on Plane Orientation

If the camera translation direction is not exactly parallel to the optical axis, images are in need of rectification, or the incline angles have to be considered in computing the c-values.

Let us consider the effect of an error in the rectification process. A similar but somewhat simpler phenomenon holds when the planes considered are not exactly parallel or orthogonal to one of the axis of the coordinate system. Again, a false association ($\widetilde{c}$) between a c-value and $\|\mathbf{w}\|$ is made. Table 4 lists the effects of every rotation (about X, Y or Z) on the c-values of each model (a: road, b: building, c: obstacle). In this table, $\widetilde{c}$-values are obtained from the new expression of velocities after rotation (see Appendix D, Tables 5, 6 and 7) when considering that the error introduced by the misalignment is residual and θ is small.

To give an idea of the phenomena, the case "rotation/X of plane model (b)" or symmetrically "rotation/Y of plane model (a)" is examined in more details. Indeed, perturbations give similar effects for all planes and rotations, but this case leads to the simplest computations. It is illustrated by Fig. 14(a) in the theoretical image plane (a 3° rotation), by

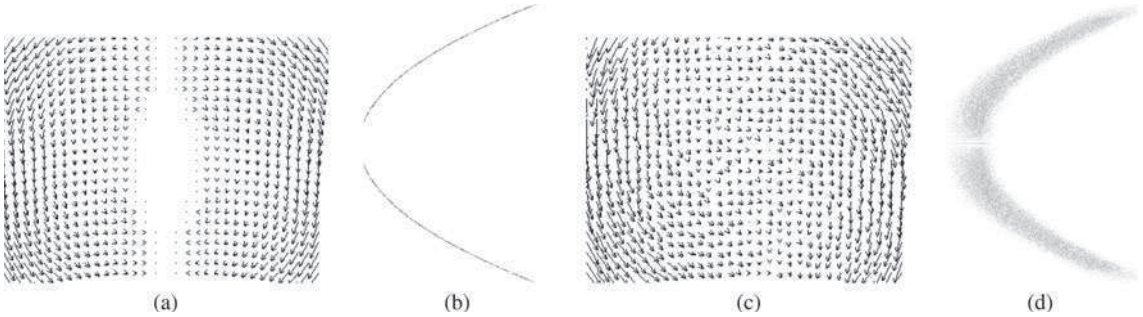


Fig. 12 Optical flow imprecision. (**a**) Original flow image. (**b**) The corresponding *c-velocity* space. (**c**) A stochastic noise is added to velocities. (**d**) Noisy *c-velocity* space. The parabola is now thicker according to noise variance

Table 4 Error on c-values in the case of rotations

	Rotation /X	Rotation /Y	Rotation /Z
(a)	$\widetilde{c}^2 = (y + f\theta)^4 + x^2(y + f\theta)^2$	$\widetilde{c}^2 = y^2(x + f\theta)^2 + y^4$	$\widetilde{c}^2 = x^2(y - x\theta)^2 + y^2(y - x\theta)^2$
(b)	$\widetilde{c}^2 = x^2(y + f\theta)^2 + x^4$	$\widetilde{c}^2 = (x + f\theta)^4 + y^2(x + f\theta)^2$	$\widetilde{c}^2 = x^2(x + y\theta)^2 + y^2(x + y\theta)^2$

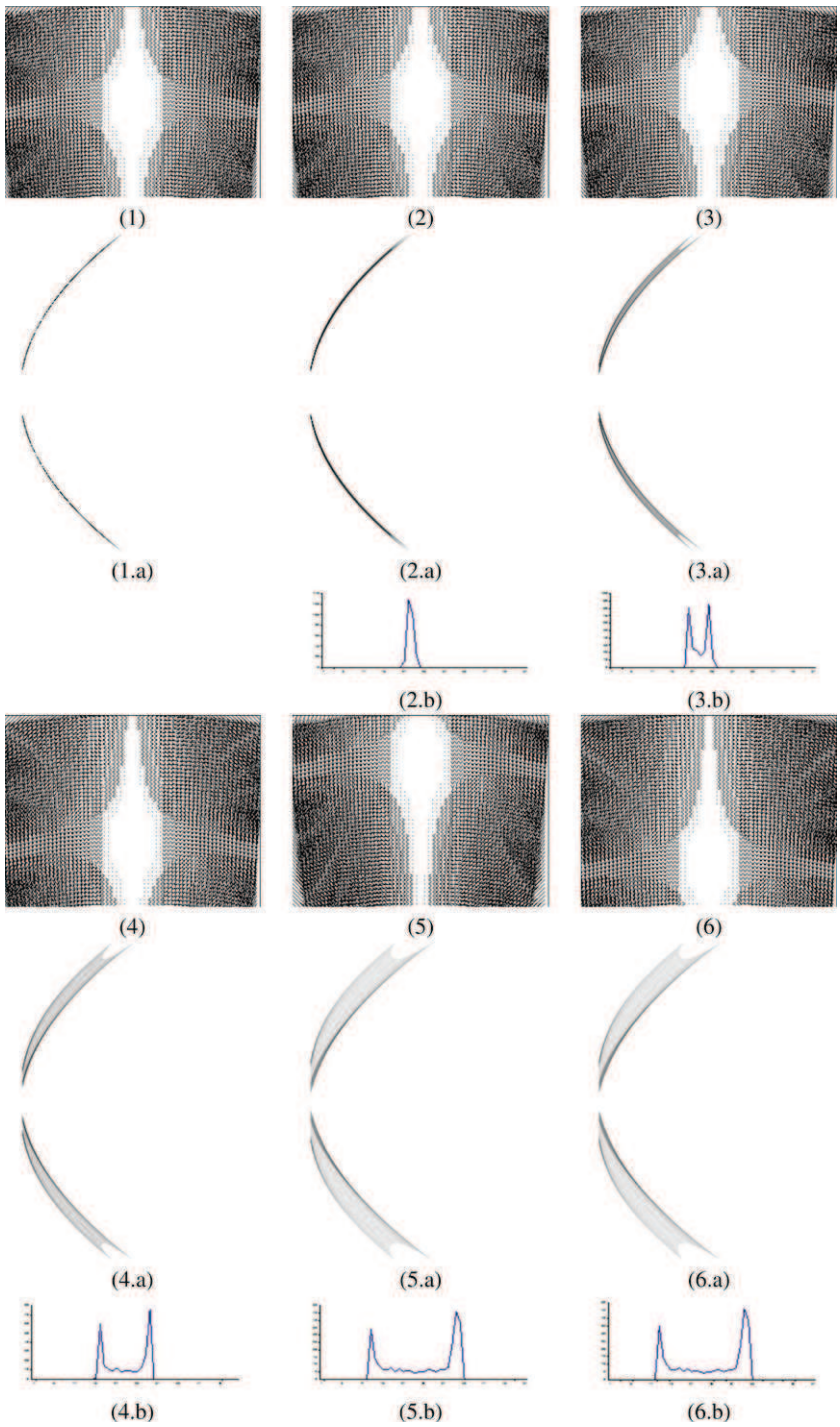


Fig. 13 Simulated building flow under increasing camera rotation about X, by 0°(**1**), 0.5° (**2**), 1.5° (**3**), 3° (**4**), 6° (**5**) and −6° (**6**). Corresponding $(\sqrt{c}, \|\mathbf{w}\|)$ frames, images (**a**). Line profiles in the $(\sqrt{c}, \|\mathbf{w}\|)$ frame at $\sqrt{c} = 70$ (∼third of the image height), images (**b**)

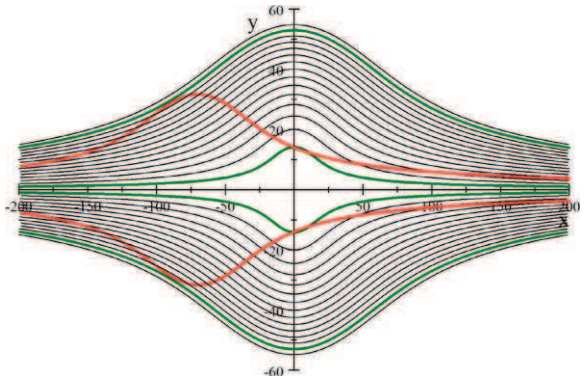


Fig. 14 (Color online) Given a $\widetilde{c}$ curve in *red*, extracted from a shifted version of the c-curve family, theoretical limits in *green* of crossing c-curves

Fig. 13(1) to (5) in the simulated image velocity field (rotations 0.5° to 6°), and by Fig. 13(1.a) to (5.a) in the resulting $(\sqrt{c}, \|\mathbf{w}\|)$ decision space. One can prove that (Bouchafa and Zavidovique 2010), whatever the translation, $a = f\theta$:

- First, the number of intersections between a translated curve $\widetilde{c}$—where $\|\mathbf{w}\|$ is actually constant—and the family of c-curves with axis $y = 0$—where $\|\mathbf{w}\|$ is practically searched for—is bounded, which is obvious from Fig. 14.
- Second, knowing that intersections vanish beyond bounds, it is clear that the corresponding limit c-curves are tangent to $\widetilde{c}$. This fact can be used to show the relation between $\widetilde{c}$ (i.e., $\|\mathbf{w}\|$) and $\sqrt{c}$ in this limit situation.

However, the corresponding effect is not predominant, mainly due to quantization. Indeed the number of voters in the contact zone between c and $\widetilde{c}$ ranges from two to a dozen, while the merging of the asymptotic parts of curves, as displayed Fig. 14, leads to a number of voters that ranges in the few dozens, possibly up to the dimension of the corresponding plane. It is shown in Bouchafa and Zavidovique (2010) how the 3 curves $\sqrt{c}$, $\sqrt{\widetilde{c}}$ and $\sqrt{c}+\sqrt{a}$ are mixed up for $x \geq \sqrt{\frac{c}{a}}$. Eventually, one can check from Fig. 13 that:

- The number of voters spreads increasingly with a, i.e., the distance between limit curves for constant $\|\mathbf{w}\|$ or constant $\sqrt{c}$ grows accordingly with a (i.e., with the rotation θ).
- A maximum number of voters belongs to the limit curves and the phenomenon is symmetric for positive and negative a corresponding to the simulated flow of a vertical plane lying over the whole image.

6.4 FOE Shift

A shift of the FOE results primarily in a relative shift of the c-curves and thus in perturbations that are analogous to the ones above. It is all the more true as it is assumed here again that the FOE position is served (Fig. 3) with respect to its theoretical setting at the picture center, and thus discrepancies are low. For that type of error, one can compute the uncertainty. Knowing that the uncertainty to be evaluated is on K where $K = \frac{\|\mathbf{w}\|}{c}$ and separating the uncertainty on $\|\mathbf{w}\|$ that is independent because it is bound to the optical flow estimation, it is clear that

$$\frac{\Delta K}{K} = \frac{\Delta c}{c} \tag{9}$$

Because

$$c = \begin{cases} \rho \\ \rho x \\ \rho y \end{cases} \quad \text{where } \rho = \sqrt{x^2 + y^2}$$

depending on the chosen model, and the relative uncertainty can be estimated through log calculation, then for each case:

$$\frac{\Delta c}{c} = \begin{cases} \frac{\Delta\rho}{\rho} = |\frac{1}{x^2+y^2}|(\Delta x + \Delta y) \\ \frac{\Delta x}{x} + \frac{\Delta\rho}{\rho} = |\frac{1}{x} + \frac{1}{x^2+y^2}|(\Delta x + \Delta y) \\ \frac{\Delta y}{y} + \frac{\Delta\rho}{\rho} = |\frac{1}{y} + \frac{1}{x^2+y^2}|(\Delta x + \Delta y) \end{cases}$$

The largest values of uncertainty are obtained for small x and y values (near the center of the image).

6.5 Inter-model Perturbation

Each point in the image gives its contribution through the cumulative process on each voting space, regardless of the model (obstacle, building, road) to which it belongs. Through that bias, the process introduces some inter-model perturbation. Of course, each contribution to other models is expected to be negligible; however, the inter model perturbation rate has to be computed more precisely. Let us distinguish between curve parameters c_b, c_r and c_o for, respectively, the building, the road and the obstacle voting spaces (in fact, we deal with $\varsigma_b = \sqrt{c_b}$, $\varsigma_r = \sqrt{c_r}$ and c_o). Let us consider now a building that verifies $\|\mathbf{w}_b\| = K_b(\sqrt{c_b})^2 = K_b\varsigma_b^2$. Representing this building in the road voting space means that a false association is made between $\|\mathbf{w}_b\|$ (derived from the optical flow) and ς_r. The relation between c_b and c_r is easily determined by $(\frac{c_r}{c_b} = |\frac{y}{x}|)$, which means that the building is represented in the road voting space by a family of parabolas with the parameter $K_b\alpha$, for $\alpha = |\frac{x}{y}|$ varying inside the aperture angle of the building plane (Bouchafa and Zavidovique 2008). A similar phenomenon holds for the other models. The equations below sum up the relations in every couple of models (only the fleeing/approaching obstacle model is considered here):

$$\frac{c_r}{c_b} = \left|\frac{y}{x}\right|, \qquad \frac{c_r}{c_{of}} = |y|, \qquad \frac{c_b}{c_{of}} = |x|$$

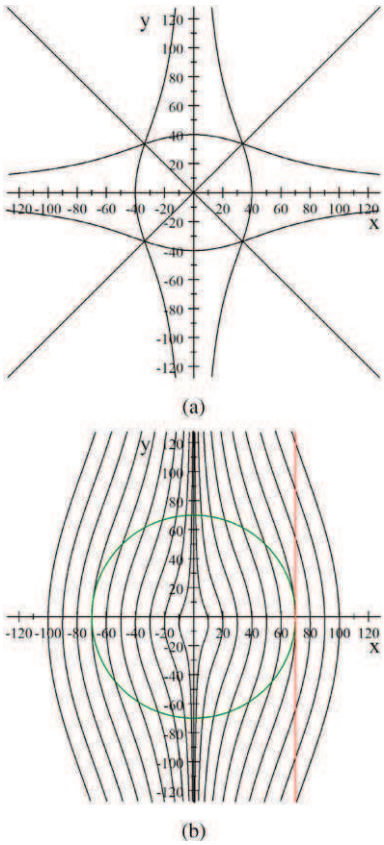


Fig. 15 (**a**) The number of increments in the cell $(\varsigma_r, \|\mathbf{w}_b\|)$ is the number of intersection points between the 3 displayed curves. (**b**) The number of increments in the cell $(c_o, \|\mathbf{w}_b\|)$ is the number of intersection points between the 3 displayed curves

For the sake of illustration, three cases are further outlined below, assuming there is one building in the image. What is the projection of this building in the road and obstacle spaces or the projection of the obstacle in the building space, and what is the maximum cumulative value to be expected?

6.5.1 *A Building Plane Is Projected on the Road c-velocity Space*

Given a parabola extracted from the *family of parabolas* explained above, corresponding to the fixed $\alpha = |\frac{x}{y}|$ value, the theoretical number of increments in the cell $(\varsigma_r, \|\mathbf{w}_b\|)$ is the number of intersection points between the 3 curves:

- $c_b = \sqrt{x^4 + x^2y^2}$, where points with velocity $\mathbf{w}_b$ are located.
- $c_r = \sqrt{y^4 + x^2y^2}$, where points with velocity $\mathbf{w}_b$ are searched.
- $|\frac{x}{y}| = \alpha$, where the $K_b\alpha$ proportionality holds.

Four intersection points are shown in Fig. 15(a), two intersection points on both sides of the center of the image. It means that the local contribution of a building to the road space is very low, as shown in Fig. 16. Nevertheless, it should be remembered that

- The thickness of the curve is likely more than one (Fig. 11) and variable, thus increasing the cell content.
- All intersections along the $|\frac{x}{y}| = \alpha$ lines contribute in the 1D-Hough space to put the $K_b\alpha$ parabola forward. The count of the corresponding cell is then at least equal to the plane dimension.

6.5.2 *A Building Plane is Projected on the Fleeing Obstacle c-Velocity Space*

The building verifies $\|\mathbf{w}_b\| = K_b c_b = K_b \varsigma_b^2$. Likewise, it is represented in the obstacle voting space by *a family of parabolas* with parameter $K_b|x|$ (i.e. $\|\mathbf{w}_b\| = K_b|x|c_o$). The number of increments in the cell $(c_o, \|\mathbf{w}_b\|)$ is the number of intersection points between the 3 curves:

$$\begin{cases} c_b = \sqrt{x^4 + x^2y^2} \\ c_o = \sqrt{x^2 + y^2} \\ \alpha = |x| \end{cases}$$

Unlike in the former case where curve families are orthogonal, the circle, the parabola and the line can be tangent (see Fig. 15b). In this case, depending on the discretization process, the number of cell increments grows significantly.

6.5.3 *An Obstacle is Projected on the Building Space*

The obstacle verifies $\|\mathbf{w}_o\| = K_o c_o$. Because $\frac{c_b}{c_o} = |x|$, the building is represented in the obstacle voting space by *a family of lines* with parameter $K_o \frac{1}{|x|}$. The number of increments in the cell $(c_o, \|\mathbf{w}_b\|)$ is as in the previous case, the number of intersection points between the 3 curves:

$$\begin{cases} c_b = \sqrt{x^4 + x^2y^2} \\ c_o = \sqrt{x^2 + y^2} \\ \alpha = \frac{1}{|x|} \end{cases}$$

Remark Figure 15 stresses again the final remark of Sect. 4 concerning both Sect. 6.5.2 and Sect. 6.5.3. Indeed, the circle where $\|\mathbf{w}\|$ is constant is tangent to the external c_b-curve and intersects some others before, which generates between a few dozen to a few voters in favor of the same modified K_o. Now, considering that frontal obstacles, fleeing or crossing, are relatively small (e.g., car, cyclist or pedestrian in front, or side-wall upper part of a building) and respective to the 9-pixels window size of the optical flow, very few $\|\mathbf{w}\|$ values are generated as a few contiguous straight segments.

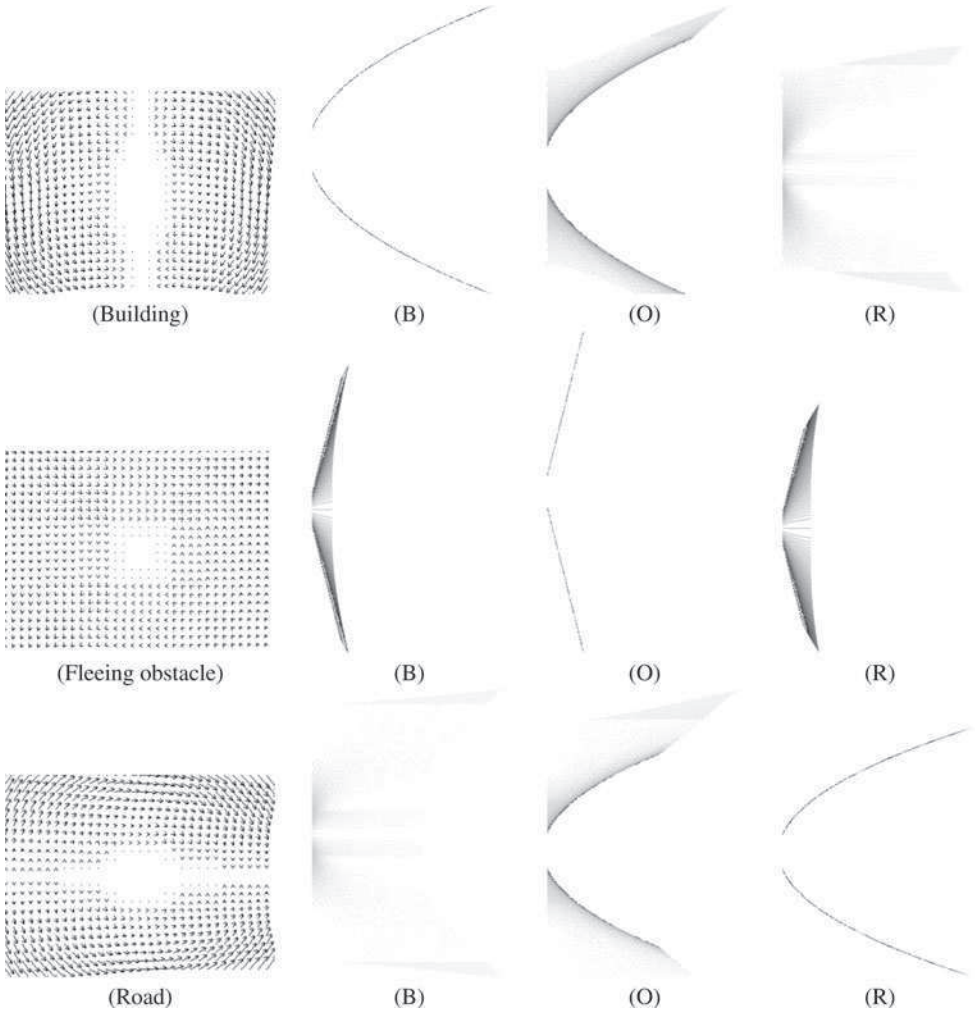


Fig. 16 Various synthetic flows and corresponding *c-velocity* spaces. (*B*): Building, (*O*): Obstacle and (*R*): Road voting spaces

6.5.4 Sum Up and Examples

In Fig. 16, a synthetic 2D motion field for each case was generated: building, obstacle and road. For each image, the corresponding *c-velocity* spaces are shown: building (B), obstacle (O) then road (R) voting space. The interpretation is straight forward in light of the explanations above. Let us still insist on the order of values. Given c, a building or road in the obstacle frame gets the maximum count for the maximum $\mathbf{w}$ and a null count beyond. It fits perfectly with Fig. 15, where, after the circle c is tangent to a given c_b-curve (resp. c_r-curve) in the family, no other c_b-curve (resp. c_r-curve) intersects, and before that, the decreasing c_b (resp. c_r) and then $\|\mathbf{w}\|$ leads to smaller and smaller counts. Likewise, the interference of road/building leads to even more clear spread because the respective curve families are orthogonal. Note that the dissymmetry r/b vs. b/r comes from the different horizontal and vertical image dimensions. This effect can be noticed again in the obstacle frame where the c extension is larger in x (building) than in y (road). Eventually, in this same frame, it is interesting to underline again that, symmetric to the cases "obstacle into road or building frames", the maximum counts are obtained for minimum $\|\mathbf{w}\|$, with the circle tangent to the curve, and the count is null before and decreases beyond with intersections.

7 Conclusion

This article shows results of a vision process to analyze a 3D scene from one moving camera. The 3D environment is assimilated into a set of planes, both vertical ones (orthogonal or parallel to motion) and horizontal ones. The approximation is efficient enough to support navigation in a urban environment: it can eliminate the depth variable, thus exhibiting a theoretical proportionality between the perceived velocity, $\|\mathbf{w}\|$, and the so-called *c-velocity*, c, i.e., curves along which a plane velocity is constant. Reciprocally, the latter linear relation characterizes planes under study. Despite its

fundamental simplicity—a Hough transform in the $(c, \|\mathbf{w}\|)$ space—and approximations made to it, the novel process proves surprisingly robust over the dozen sequences, each of several thousand images, that were used in the experiments. It detects more than 80% correct planes on average with no misses on frontal planes along sequence parts with straight moves. The good results require further analysis of the main perturbations bound to the method. An explanation was offered for why the spread of theoretical parabolas into "paraboloidal" clusters of $(\sqrt{c}, \|\mathbf{w}\|)$ cells does not jeopardize the detection. It was shown how:

- The parabolas expand within parabolic bounds due to camera or plane misalignment.
- The inter-model contamination distributes the same curves over a bounded family of parabolas due to projections along rays in the image space.
- The *moiré* effect due to curve displacements impacts only the vicinity of the FOE where the velocity is weak anyway, which contributes to the stability as well.
- The sub-sampling imposed by very large integer numbers involved—in the order of the image dimension to the power four—ends contributing positively despite the variable curve increment it produces, in increasing the number of voting cells per parabola in the Hough space.

Considering the simplicity of its real-time implementation, the process could support generalization to limited rotation towards more realistic motion control-wise. Along the same line, should applications require it, the generalization to 3D patterns as cones or cylinders seems sensible, subject however to the computing and acquisition accuracy.

Acknowledgements This work was undertaken in the framework of the French ANR project LOVe (pedestrian detection using an on-board camera for driver assistance). We would like to thank Antoine Patri, a research engineer, for his contribution to the implementation of the algorithms on the RTMaps software platform.

Appendix A: Two-dimensional velocity (u, v) from three-dimensional motion T, Ω

Consider a coordinate system $OXYZ$ at the optical center of a pinhole camera, such that the axis OZ coincides with the optical axis (see Fig. 17).

In the case of rigid motion, when a camera moves with a translational instantaneous velocity $\mathbf{T} = (T_X, T_Y, T_Z)$ and a rotational instantaneous velocity $\Omega = (\Omega_X, \Omega_Y, \Omega_Z)$, each static scene point $\mathbf{P} = (X, Y, Z)$ moves relative to the camera with a velocity $\mathbf{V}$ given by

$$\mathbf{V} = \frac{d\mathbf{P}}{dt} = \begin{pmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{pmatrix} = -\mathbf{T} - \Omega \times \mathbf{P} \tag{10}$$

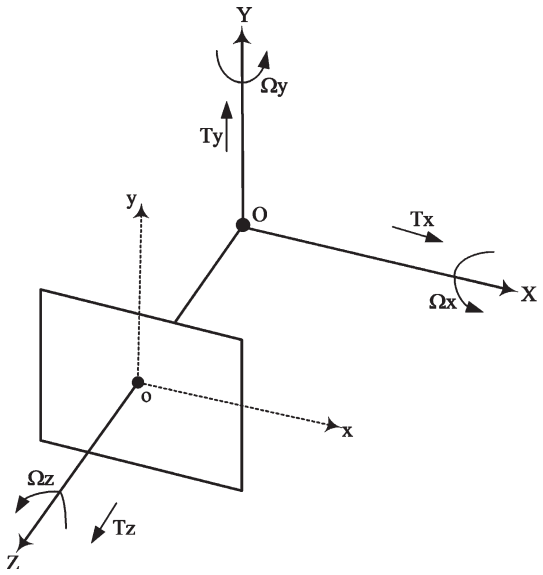


Fig. 17 The coordinate system considered assuming a pinhole moving camera model

$$\Longleftrightarrow \quad \begin{cases} \dot{X} = -T_X - \Omega_Y Z + \Omega_Z Y \\ \dot{Y} = -T_Y - \Omega_Z X + \Omega_X Z \\ \dot{Z} = -T_Z - \Omega_X Y + \Omega_Y X \end{cases} \tag{11}$$

The projection of the point $\mathbf{P} = (X, Y, Z)$ in the image plane is

$$p = \begin{pmatrix} x \\ y \end{pmatrix} = f \begin{pmatrix} \frac{X}{Z} \\ \frac{Y}{Z} \end{pmatrix} \tag{12}$$

The derivation of the previous expression gives the 2D velocity $\mathbf{v} = (\dot{x}, \dot{y})$ of each image point $\mathbf{p} = (x, y)$:

$$\begin{aligned} \dot{x} &= \frac{\dot{X}Z - \dot{Z}X}{Z^2} \\ \dot{y} &= \frac{\dot{Y}Z - \dot{Z}Y}{Z^2} \end{aligned} \tag{13}$$

Substituting (11) and (12), the 2D velocity vector for each image point is

$$\begin{cases} \dot{x} = \left(-f\dfrac{T_X}{Z} - \Omega_Y + y\Omega_Z\right) \\ \qquad - x\left(-\dfrac{T_Z}{Z} - y\dfrac{\Omega_X}{f} + x\dfrac{\Omega_Y}{f}\right) \\ \dot{y} = \left(-f\dfrac{T_Y}{Z} - x\Omega_Z + \Omega_X\right) \\ \qquad - y\left(-\dfrac{T_Z}{Z} - y\dfrac{\Omega_X}{f} + x\dfrac{\Omega_Y}{f}\right) \end{cases}$$

Rearranging the terms to group rotations and translations gives ($u = \dot{x}$, $v = \dot{y}$):

$$\begin{cases} u = \dfrac{xy}{f}\Omega_X - \left(\dfrac{x^2}{f} + 1\right)\Omega_Y + y\Omega_Z - \dfrac{fT_X + xT_Z}{Z} \\ v = -\dfrac{xy}{f}\Omega_Y - \left(\dfrac{y^2}{f} + 1\right)\Omega_X - x\Omega_Z - \dfrac{fT_Y + yT_Z}{Z} \end{cases} \tag{14}$$

Appendix B: Two-Dimensional Velocity of a Moving Plane

Suppose now that the camera is observing a planar surface of equation $\mathbf{n}^T\mathbf{P} = d$, with $\mathbf{n} = (n_X, n_Y, n_Z)$ the unit vector normal to the plane, d the distance "plane to origin" and P the generic point (X, Y, Z). After (Longuet-Higgins and Prazdny 1980; Verri and Poggio 1989) or from (14) and $Z = \frac{1}{n_Z}(d - n_X X - n_Y Y)$, the 2D velocity can be written as

$$\begin{cases} u = \dfrac{1}{fd}(a_1x^2 + a_2xy + a_3fx + a_4fy + a_5f^2) \\ v = \dfrac{1}{fd}(a_1xy + a_2y^2 + a_6fy + a_7fx + a_8f^2) \end{cases} \tag{15}$$

$$\begin{aligned} a_1 &= -d\Omega_Y + T_Z n_X \\ a_2 &= d\Omega_X + T_Z n_Y \\ a_3 &= T_Z n_Z - T_X n_X \\ a_4 &= d\Omega_Z - T_X n_Y \\ a_5 &= -d\Omega_Y - T_X n_Z \\ a_6 &= T_Z n_Z - T_Y n_Y \\ a_7 &= -d\Omega_Z - T_Y n_X \\ a_8 &= d\Omega_X - T_Y n_Z \end{aligned}$$

Appendix C: Cumulative Curve Rectification

Case 1: road model, computing c with respect to y In this case, $c^2 = y^2(x^2 + y^2)$. A relation between c and y requires the solution of $y^4 + y^2x^2 - c^2 = 0$. Hence,

$$y = \mp\sqrt{\frac{-x^2 + \sqrt{x^4 + 4c^2}}{2}} \tag{16}$$

Case 2: building model, computing c with respect to x In this case, $c^2 = x^2(x^2 + y^2)$. A relation between c and x requires the solution of $x^4 + y^2x^2 - c^2 = 0$. Hence,

$$x = \mp\sqrt{\frac{-y^2 + \sqrt{y^4 + 4c^2}}{2}} \tag{17}$$

Case 3: fleeing/approaching obstacle, computing c with respect to y In this case, $c^2 = x^2 + y^2$. The relation between c and y is trivially

$$y = \mp\sqrt{c^2 - x^2} \tag{18}$$

Appendix D: Errors on Velocity Vectors in the Case of Camera Rotation

Table 5 Error on the velocity vectors in the case of a rotation/X

	$\mathbf{n}$	Rotation/X $\mathbf{T}' = (0, -\sin\theta, \cos\theta)T_Z$
(a)	$(0, 1, 0)$	$\mathbf{n}' = (0, \cos\theta, \sin\theta)$ $\begin{cases} u = K[x\cos\theta(y\cos\theta + f\sin\theta)] \\ v = K(y\cos\theta + f\sin\theta)^2 \end{cases}$
(b)	$(1, 0, 0)$	$\mathbf{n}' = (1, 0, 0)$ $\begin{cases} u = Kx^2\cos\theta \\ v = K[x(y\cos\theta + f\sin\theta)] \end{cases}$
(c)	$(0, 0, 1)$	$\mathbf{n}' = (0, -\sin\theta, \cos\theta)$ $\begin{cases} u = K[x\cos\theta(f\cos\theta - y\sin\theta)] \\ v = K[(y\cos\theta + f\sin\theta)(f\cos\theta - y\sin\theta)] \end{cases}$

Table 6 Error on the velocity vectors in the case of a rotation/Y

	Rotation/Y $\mathbf{T}' = (-\sin\theta, 0, \cos\theta)T_Z$
(a)	$\mathbf{n}' = (0, 1, 0)$ $\begin{cases} u = Ky(x\cos\theta + f\sin\theta) \\ v = Ky^2\cos\theta \end{cases}$
(b)	$\mathbf{n}' = (\cos\theta, 0, \sin\theta)$ $\begin{cases} u = K(x\cos\theta + f\sin\theta)^2 \\ v = K[y\cos\theta(x\cos\theta + f\sin\theta)] \end{cases}$
(c)	$\mathbf{n}' = (-\sin\theta, 0, \cos\theta)$ $\begin{cases} u = K(f\cos\theta - x\sin\theta)(x\cos\theta + f\sin\theta) \\ v = Ky\cos\theta(f\cos\theta - x\sin\theta) \end{cases}$

Table 7 Error on the velocity vectors in the case of a rotation/Z

	Rotation/Z $\mathbf{T}' = \mathbf{T}$
(a)	$\mathbf{n}' = (-\sin\theta, \cos\theta, 0)$ $\begin{cases} u = K[x(-x\sin\theta + y\cos\theta)] \\ v = K[y(-x\sin\theta + y\cos\theta)] \end{cases}$
(b)	$\mathbf{n}' = (\cos\theta, \sin\theta, 0)$ $\begin{cases} u = K[x(x\cos\theta + y\sin\theta)] \\ v = K[y(x\cos\theta + y\sin\theta)] \end{cases}$
(c)	$\mathbf{n}' = (0, 0, 1)$ $\begin{cases} u = Kfx \\ v = Kfy \end{cases}$

References

Bay, H., Ess, A., Tuytelaars, T., & Gool, L. V. (2008). Surf: speeded up robust features. *Computer Vision and Image Understanding*, *110*(3), 346–359.

Bouchafa, S., & Zavidovique, B. (2006). Efficient cumulative matching for image registration. *Image and Vision Computing*, *24*(1), 70–79.

Bouchafa, S., & Zavidovique, B. (2008). Moving plane detection under translational camera motion using the c-velocity concept. In *IEEE international workshop on image processing, theory tools and applications* (pp. 1–8).

Bouchafa, S., & Zavidovique, B. (2010). Robustness of c-velocity based methods for 3d moving plane detection. In *Proc. of 10th international conference on pattern recognition and image analysis: new information technologies (PRIA)*, St. Petersburg, Russia (pp. 177–181).

Duda, R., & Hart, P.E. (1972). Use of the hough transformation to detect lines and curves in pictures. *Communications of the ACM*, *15*, 11–15.

Fermuller, C., & Aloimonos, Y. (1995). Global rigidity constraints in image displacement fields. In *International conference on computer vision* (pp. 245–250).

Hanes, D., Keller, J., & McCollum, G. (2008). Motion parallax contribution to perception of self-motion and depth. *Biological Cybernetics*, *98*(4), 273–293.

Hartley, R. (1995). In defense of the 8-point algorithm. In *IEEE international conference on computer vision* (pp. 1064–1070).

Hildreth, E. C. (1992). Recovering heading for visually-guided navigation. *Vision Research*, *32*(6), 1177–1192.

Irani, M., Rousso, B., & Peleg, S. (1997). Recovery of egomotion using region alignment. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *19*(3), 268–272.

Labayrade, R., Aubert, D., & Tarel, J. (2002). Real time obstacle detection on non flat road geometry through 'v-disparity' representation. In *IEEE intelligent vehicles symposium 2002* (pp. 646–651).

Longuet-Higgins, H., & Prazdny, K. (1980). The interpretation of a moving retinal image. *Proceeding of Royal Society London*, *B-208*, 385–397.

Lucas, B., & Kanade, T. (1981). An iterative image registration technique with an application to stereovision. In *DARPA image understanding workshop* (pp. 121–130).

Luong, Q. T., & Faugeras, OD (1997). Camera calibration, scene motion and structure recovery from point correspondences and fundamental matrices. *International Journal of Computer Vision*, *22*(3), 261–289.

MacKay, D. J. (2003). *Information theory, inference, and learning algorithms*.

MacLean, W., Jepson, A., & Frecker, R. (1994). Recovery of egomotion and segmentation of independant object motion using the em algorithme. In *British machine vision conference* (pp. 13–16).

Negahdaripour, S., & Horn, B. (1989). A direct method for locating the focus of expansion. *Computer Vision, Graphics, and Image Processing*, *46*(3), 303–326.

Roberts, R., Potthast, C., & Dellaert, F. (2009). Learning general optical flow subspaces for egomotion estimation and detection of motion anomalies. In *IEEE conference on computer vision and pattern recognition*, Miami, USA (pp. 57–64).

Sazbon, D., Rotstein, H., & Rivlin, E. (2004). Finding the focus of expansion and estimating range using optical flow images and a matched filter. *Machine Vision and Applications*, *15*(4), 229–236.

Stein, G. P., Mano, O., & Shashua, A. (2000). A robust method for computing vehicle egomotion. In *IEEE intelligent vehicles symposium* (pp. 362–368).

Verri, A., & Poggio, T. (1989). Motion field and optical flow: qualitative properties. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *11*(5), 490–498.

2010 IEEE Intelligent Vehicles Symposium
University of California, San Diego, CA, USA
June 21-24, 2010

Detection of Independently Moving Objects Through Stereo Vision and Ego-Motion Extraction

Adrien Bak, Samia Bouchafa
UniverSud, Universite Paris XI
Institut d'Electronique Fondamentale
Orsay, France
Email : firstname.name@ief.u-psud.fr

Didier Aubert
UniverSud, LIVIC
INRETS, LCPC
Versailles, France
Email : firstname.name@inrets.fr

Abstract— **Vision-based autonomous vehicles must face numerous challenges in order to be effective in practical areas. Among these lies the detection and localization of independent-moving objects, so as to track or avoid them. In this paper a method that address this particular issue is presented. Information from stereo and motion is used to extract the ego-motion of the vehicle. Known defects of this estimation are exploited to detect independent-moving obstacles. This method allows an early and reliable detection, even for objects partially occluded. Besides, it highlights the errors in the disparity map, which can be used, in future works, to correct depth-estimation, through motion-estimation.**

I. Introduction

In order to develop an independent, mobile robot, one must first take a glance at the obstacle detection. The work described in this paper addresses such an issue. To that end, only visual information is to be used. Modern cars can be equipped with a variety of sensors (like GPS, proprioceptive sensors, collision detectors, *etc.*) but those sensors are.On the contrary, vision provides a much richer data and can serve several purposes, such as (but not limited to) localization, recognition and pathfinding. Anthropological and psycho-cognitive evidence [1] shows us the importance of visual information in human motivity and development. As such, computer vision, applied to intelligent vehicles is a highly active research topic, one can refer to [2] for a more extensive overview of the topic.

One can basically distinguish between monocular and binocular approaches. Monocular approaches, such as [3], [4] rely on image motion estimation, through the computation of optical flow [5]. Some authors, such as [6], estimate the motion directly from the image, but they still have to rely on the *brightness constraint equation*. However, the ill-posed nature of the optical flow computation problem makes the use of regularization or heavy smoothing constraints [7], [8] necessary. Such constraints can deteriorate the useful information in image region such as occlusion regions, or depth-discontinuities. Besides, monocular methods lack the exact knowledge of objects depth and can only determine the exact position of a given object up to a scale factor. On the other hand stereovision based methods provide, through calibration, an absolute measurement of a 3D space. Disparity information can be used in order to detect, without any other input, potential obstacles [9], [10].

The information provided by both cues is complementary, thus a current trend is to make those collaborate, in order to exploit motion analysis and scene structure. For instance, the past decade has seen many attempts to achieve a useful collaboration in the domain of obstacle detection [11], [12] or in the field of ego-motion recovery (odometry) and pathfinding [13]–[15].

Some authors, such as [16] have tried to estimate the ego-motion of a stereo-rig and then compute a 3D-displacement field due to this ego-motion, in order to identify dynamic objects. However their method differs from the one described here on several points. First, they use the predicted displacement field only to discriminate between static and dynamic objects, stereo-vision is then used to extract the different targets. This could lead to detection errors, for instance, two distinct objects with different motions but with the same disparity would be merged. Moreover, their method relies on a thresholding, whereas the proposed algorithm stems from a much more robust error analysis of the most important part of such an algorithm : the ego-motion extraction. Finally, the proposed method relies on robust feature points, which are insensitive to the aperture problem and allow for greater displacement than the correlation-based optical flow proposed in [16].

The method presented here stems from [13] work on d-motion estimation. According to the error model developed in section IV, the consistency of every point with the extracted ego-motion is checked through the use of a robust correlation technique. The proposed method does not only perform a static/dynamic detection, it allows a fine segmentation with respect to the obstacle's own ego-motion. As shown in section V, an early detection can be achieved even in hard cases (occlusion for instance).

In the first section, the motion model and the hypothesis will be described. In the second section, an ego-motion extraction algorithm will be described and tested. In section IV, a method to detect independent motion in a sequence of image pair is presented. The results of this method and their discussion is to be found in section V. Finally, section VI concludes and presents some improvement that can be brought to the current method.

978-1-4244-7868-2/10/$26.00 ©2010 IEEE

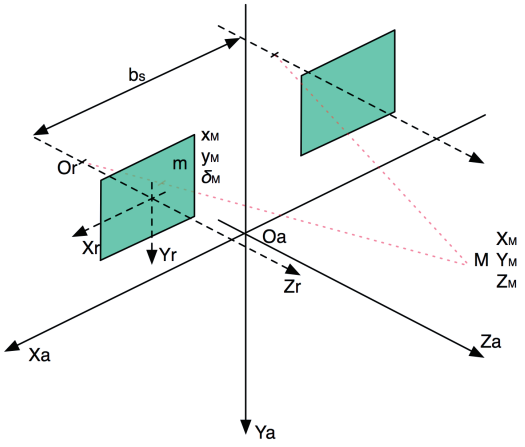


Fig. 1. Coordinate system to be used, at time $(t + \partial t)$

II. Model and Hypothesis

A mobile vehicle (*e.g.* a car), moving in a world constituted of either static or dynamic objects, is considered. The world is described by a set of two frames of reference. The first one, labelled $\mathcal{R}_a$ is absolute while the second one $\mathcal{R}_r$ is bound to the vehicle, with its origin located at the optical center of the right-hand-side vision sensor.

This vehicle is equipped with a rectified stereo rig, looking forward. The two image sensors are modeled by the pinhole camera model. Both focal lengths are identical and are noted f, the stereo rig's baseline is b_s, pixels are considered to be squared, with their dimension equal to t_p. Disparity is measured with respect to the right-hand-side coordinates. The choice of the algorithm used to recover sparse or dense disparity maps is left to the reader's discretion. One can refer to [17] for an extensive study of existing algorithms. The method that will be used is fully described in [10].

According to the standard pinhole model, a static world point $M = \left| \begin{array}{c} X_M(t) \\ Y_M(t) \\ Z_M(t) \end{array} \right|_{\mathcal{R}_a}$ is imaged by our system as :

$$m = \left| \begin{array}{c} x_M(t) = f\frac{X_M(t)-b_s/2}{Z_M(t)} \\ y_M(t) = f\frac{Y_M(t)}{Z_M(t)} \\ \delta_M(t) = f\frac{b_s}{Z_M(t)} \end{array} \right| \quad (1)$$

Where δ is the disparity between right and left images, and m belongs to the disparity space. One can easily show that the disparity space is a projective space as there exists a projective transformation between the homogeneous coordinates of a point in the 3D euclidean space and the homogeneous coordinates of its image by the stereo rig. This work will be conducted in the disparity space, because of the isotropic nature of the associated discretization noise as shown in [13].

Static objects are assumed to be dominant in the images. This assumption will later allow to extract the vehicle ego-motion. Without any loss of generality and unless explicit notice, only two different times, labelled t and $t + \partial t$ are considered. At t, the two coordinates systems are coincident. The motion between $\mathcal{R}_a$ and $\mathcal{R}_r$, that occurs between t and $t + \partial t$ is decomposed in its translational and rotational components.

$$\overrightarrow{T}(t) = \left| \begin{array}{c} T_X(t) \\ T_Y(t) \\ T_Z(t) \end{array} \right|_{\mathcal{R}_a}$$

$$\overrightarrow{\Omega}(t) = \left| \begin{array}{c} \omega_X(t) \\ \omega_Y(t) \\ \omega_Z(t) \end{array} \right|_{\mathcal{R}_a}$$

The angles $\omega_X(t)$, $\omega_Y(t)$ and $\omega_Z(t)$ are considered to be small enough to linearize trigonometric lines. This assumption is valid in standard driving conditions. After the motion $\left\{\overrightarrow{T}, \overrightarrow{\Omega}\right\}$, the static world point M can be expressed as:

$$M = \left| \left| \begin{array}{c} X_M(t) \\ Y_M(t) \\ Z_M(t) \end{array} \right| + \overrightarrow{\Omega}(t) \wedge \left| \begin{array}{c} X_M(t) \\ Y_M(t) \\ Z_M(t) \end{array} \right| - \overrightarrow{T}(t) \right|_{\mathcal{R}_r}$$

$$M = \left| \begin{array}{l} X_M(t) - \omega_Y(t).Z_M(t) + \omega_Z(t).Y_M(t) - T_X(t) \\ Y_M(t) + \omega_X(t).Z_M(t) - \omega_Z(t).X_M(t) - T_Y(t) \\ Z_M(t) + \omega_Y(t).X_M(t) - \omega_X(t).Y_M(t) - T_Z(t) \end{array} \right|_{\mathcal{R}_r}$$

In order to model motion in the disparity space, the so-called d-motion formalism is used. Assuming that $m(t) \left| \begin{array}{c} x(t) \\ y(t) \\ \delta(t) \end{array} \right|$ is the image of a static world point, it will be mapped as $m(t + \partial t)$. As of now, $variable(t)$ will simply be noted $variable$, and $variable(t+\partial t)$ will be noted $variable'$. The coordinates of m' can be expressed through the projections of the motion components equations:

$$m' = \left| \begin{array}{c} x' \\ y' \\ \delta' \end{array} \right| = \left| \begin{array}{c} \frac{x+\omega_Z.y-\frac{T_X.\delta}{b_s}.f-f.\omega_Y}{\frac{\omega_Y}{f}.x-\frac{\omega_X}{f}.y-\frac{T_Z.\delta}{b_s}+1} \\ \frac{y-\omega_Z.x-\frac{T_Y.\delta}{b_s}.f+f.\omega_X}{\frac{\omega_Y}{f}.x-\frac{\omega_X}{f}.y-\frac{T_Z}{f.b_s}+1} \\ \frac{\delta}{\frac{\omega_Y}{f}.x-\frac{\omega_X}{f}.y-\frac{T_Z}{f.b_s}+1} \end{array} \right| \quad (2)$$

In the following the following will be used :

$$m' = P_{\left(\overrightarrow{\Omega},\overrightarrow{T}\right)}(m)$$

III. Ego-Motion Extraction

A. Method

In order to identify moving obstacles, one must first evaluate its own ego-motion. For that we consider a set of N point correspondences $\{m_i, i \in [1, N]\} \rightarrow \{m'_i \in [1, N]\}$. Those correspondence can be provide by an optical flow method, but we'd rather rely on more robust, feature points extraction methods, such as Harris corner detector [18], the SURF detector [19], or level-lines junction [20]. Because of its trade-off between robustness and computational cost, a SURF detector will be used later on.

Equation (2) provides a linear system in $(\overrightarrow{\Omega}, \overrightarrow{T})$:

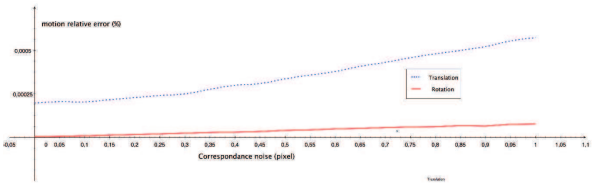


Fig. 2. Ego-motion estimation relative error Vs. correspondences noise. Used data consists of 500 points, 20% of which were outliers. The estimator used to evaluate relative error is the median of 10 000 realizations.

$$\begin{matrix}\vec{\Omega}\\ \vec{T}\end{matrix} \cdot \begin{pmatrix} -\frac{x'_1 \cdot y_1}{f} & \frac{x_1 \cdot x'_1}{f} + f & -y & \frac{\delta_1}{b_s} & 0 & \frac{\delta_1 \cdot x'_1}{b_s} \\ f - \frac{y_1 \cdot y'_1}{f} & \frac{x_1 \cdot y'_1}{f} & -x & 0 & \frac{\delta_1}{b_s} & \frac{\delta_1 \cdot y'_1}{b_s} \\ \frac{\delta'_1 \cdot y_1}{f} & -\frac{\delta'_1 \cdot x_1}{f} & 0 & 0 & 0 & \frac{\delta_1 \cdot \delta'_1}{b_s} \\ \cdots & & & & & \\ -\frac{x'_N \cdot y_N}{f} & \frac{x_N \cdot x'_N}{f} + f & -y & \frac{\delta_N}{b_s} & 0 & \frac{\delta_N \cdot x'_N}{b_s} \\ f - \frac{y_N \cdot y'_N}{f} & \frac{x_N \cdot y'_N}{f} & -x & 0 & \frac{\delta_N}{b_s} & \frac{\delta_N \cdot y'_N}{b_s} \\ \frac{\delta'_N \cdot y_N}{f} & -\frac{\delta'_N \cdot x_N}{f} & 0 & 0 & 0 & \frac{\delta_N \cdot \delta'_N}{b_s} \end{pmatrix} = \begin{pmatrix} x'_1 - x_1 \\ y'_1 - y_1 \\ \delta'_1 - \delta_1 \\ \cdots \\ x'_N - x_N \\ y'_N - y_N \\ \delta'_N - \delta_N \end{pmatrix}$$

The purpose is to find a couple $\left(\tilde{\vec{\Omega}}, \tilde{\vec{T}}\right)$[1] that minimizes the following :

$$\epsilon = \sum_{i=1}^{i=N} dist\left(m'_i, P_{\left(\tilde{\vec{\Omega}}, \tilde{\vec{T}}\right)}(m_i)\right)$$

where $dist(a, b)$ can be any of the usual topological distances, extended to the disparity space. All experiments were conducted with the euclidean distance. That minimization is performed, using a RANSAC [21] approach in order to reject outliers, along with singular value decomposition to solve the linear system at every step of the process. RANSAC parameters are set assuming a minimal proportion of inliers of one third of the extracted feature points and to ensure a 5% probability of false rejections [22].

B. Results

In order to evaluate the precision of the ego-motion extraction, we proceed by different means, first synthetic data is used. Such data is constituted by a set of static points and animated by an arbitrary motion, with perfect correspondences (or with a perfectly known noise) feeding the ego-motion extraction algorithm. Those results can be found in Fig. 2.

The Sivic simulator [23] was also used. This presents the advantage of providing pseudo-realistic image-sequences and a perfect knowledge of the vehicle ego-motion. Fig. 3 shows an example of such pseudo-realistic images. Fig. 4 presents results for a test sequence from Sivic. This sequence presents urban landscape, with moderate traffic. After 600 frames, and about 250 meters, the positioning error is 2.5 meters. The average instantaneous error is less than 2%.

Errors in the ego-motion estimation stems from various sources:

- First, one can note the fact that RANSAC process isn't always optimal. For practical applications, we are bound to set a maximum number of iterations, and we do not have the certainty that the final result is the best we can obtain, depending on our input data.
- Second, there is the discretization of the disparity space, and more generally, the positioning of the feature points used to inverse the motion model.

From these two error sources, only the later can be quantified. However, through the use of the Sivic simulator, an upper bound to the relative error was estimated around 15%.

IV. Detection

A. Ego-Motion Error Propagation

The objective of this section is to detect image points that don't validate the motion model found in section III. For that, from (2), one can write :

$$\begin{cases} \mu = \frac{xy}{f}\omega_X - \left(f + \frac{x^2}{f}\right)\omega_Y + y\omega_Z - \delta f \frac{T_X}{b_s} + \frac{x\delta T_Z}{b_s} \\ \nu = \left(f + \frac{y^2}{f}\right)\omega_X - \frac{xy}{f}\omega_Y - x\omega_Z - \delta f \frac{T_Y}{b_s} + \frac{y\delta T_Z}{b_s} \\ \xi = \delta \frac{y\omega_X - x\omega_Y + \frac{T_Z}{b_s}}{x\omega_Y - y\omega_X - \frac{T_Z}{b_s} + 1} \end{cases} \quad (3)$$

Where $\mu = x' - x$, $\nu = y' - y$ and $\xi = \delta' - \delta$. The following notation will be used :

$$\begin{vmatrix} \mu \\ \nu \\ \xi \end{vmatrix} = \Pi_{\left(\vec{\Omega}, \vec{T}\right)} \begin{pmatrix} x \\ y \\ \delta \end{pmatrix}$$

and $\Pi_{\left(\vec{\Omega}, \vec{T}\right)}$ can be called *displacement field*.

For every point m in the disparity-space at the time t, the coordinates of the point $m' = m + \Pi_{\left(\tilde{\vec{\Omega}}, \tilde{\vec{T}}\right)}(m)$ are computed. If the point m is the image of a static world point,

[1] the tilda symbol denotes an estimate

Fig. 3. Image extracted from a test sequence generated with the Sivic simulator

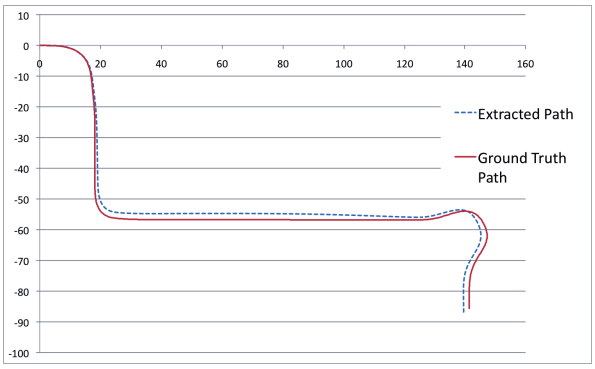


Fig. 4. Extracted and Ground Truth paths for a sequence of 600 stereo pairs.

m' should be its correspondent at the time $t+\partial t$, on the other hand, if m is the image of a world point, belonging to a dynamic object, m' shouldn't. However, as seen previously, the estimation of the ego-motion, isn't perfectly accurate. Through the propagation of uncertainty with respect to $\overrightarrow{\Omega}$ and $\overrightarrow{T}$, and with (3) :

$$\begin{cases} \partial\mu = \frac{xy}{f}\partial\omega_X + \left(f + \frac{x^2}{f}\right)\partial\omega_Y + y\partial\omega_Z - \frac{f\delta}{b_s}\partial T_X + \frac{x\delta}{b_s}\partial T_Z \\ \partial\nu = \left(f + \frac{y^2}{f}\right)\partial\omega_X - \frac{xy}{f}\partial\omega_Y - x\partial\omega_Z - \frac{f\delta}{b_s}\partial T_Y + \frac{y\delta}{b_s}\partial T_Z \\ \partial\xi = \frac{\delta}{(1-a)^2}\cdot\left(y\partial\omega_X - x\partial\omega_Y + \frac{1}{b_s}\partial T_Z\right) \end{cases} \tag{4}$$

where $a = y\omega_X - x\omega_Y + \frac{T_Z}{b_s}$

By replacing in (4) every motion-related variable by an estimate of its upper bound, the following can be defined:

$$\begin{cases} \Delta\mu = \frac{xy}{f}\Delta\omega_X + \left(f + \frac{x^2}{f}\right)\Delta\omega_Y + y\Delta\omega_Z - \frac{f\delta}{b_s}\Delta T_X + \frac{x\delta}{b_s}\Delta T_Z = P_1(x,y,\delta) \\ \Delta\nu = \left(f + \frac{y^2}{f}\right)\Delta\omega_X - \frac{xy}{f}\Delta\omega_Y - x\Delta\omega_Z - \frac{f\delta}{b_s}\Delta T_Y + \frac{y\delta}{b_s}\Delta T_Z = P_2(x,y,\delta) \\ \Delta\xi = \frac{\delta}{(1-a)^2}\cdot\left(y\Delta\omega_X - x\Delta\omega_Y + \frac{1}{b_s}\Delta T_Z\right) = P_3(x,y,\delta) \end{cases}$$

where P_1, P_2 and P_3 are polynomials. For every point, given its position in the disparity space and the current estimate of the ego-motion, the upper bounds $\Delta\mu, \Delta\nu$ and $\Delta\xi$ of the uncertainty of the displacement field can now be estimated.

With this result, for each point m of the disparity space, a 3D interval, at the time $t + \partial t$, can be defined by:

$$\begin{aligned} \mathcal{W}^m_{\left(\tilde{\overrightarrow{\Omega}},\tilde{\overrightarrow{T}}\right)} &= [x+\mu-\Delta\mu; x+\mu+\Delta\mu] \\ &\times [y+\nu-\Delta\nu; y+\nu+\Delta\nu] \\ &\times [\delta+\xi-\Delta\xi; \delta+\xi+\Delta\xi] \end{aligned}$$

where "$\times$" stands for the cartesian products.

So, for every point m in the disparity space that is the image of a static world point, it is known that :

$$\begin{aligned} &\exists\ m' \in \mathcal{W}^m_{\left(\tilde{\overrightarrow{\Omega}},\tilde{\overrightarrow{T}}\right)} \text{ such as} \\ &m' = m + \Pi_{\left(\overrightarrow{\Omega},\overrightarrow{T}\right)}(m) \end{aligned}$$

By searching $\mathcal{W}^m_{\left(\tilde{\overrightarrow{\Omega}},\tilde{\overrightarrow{T}}\right)}$ for the correspondent of m, it can be determined whether m is a static point (*i.e.* there is actually a correspondent to m in $\mathcal{W}^m_{\left(\tilde{\overrightarrow{\Omega}},\tilde{\overrightarrow{T}}\right)}$), or if m moves independently (*i.e.* no correspondent can be found within the boundaries of $\mathcal{W}^m_{\left(\tilde{\overrightarrow{\Omega}},\tilde{\overrightarrow{T}}\right)}$). The only significant limitation at this point is the case of a dynamic point d, whose motion $\left(\overrightarrow{\Omega_d},\overrightarrow{T_d}\right)$ satisfies the following :

$$\begin{aligned} \overrightarrow{\Omega_d} &\in \left[\tilde{\overrightarrow{\Omega}} - \partial\overrightarrow{\Omega}\ ,\ \tilde{\overrightarrow{\Omega}} + \partial\overrightarrow{\Omega}\right] \\ \overrightarrow{T_d} &\in \left[\tilde{\overrightarrow{T}} - \partial\overrightarrow{T}\ ,\ \tilde{\overrightarrow{T}} + \partial\overrightarrow{T}\right] \end{aligned}$$

such a point would have its correspondent within the boundaries of $\mathcal{W}^d_{\left(\tilde{\overrightarrow{\Omega}},\tilde{\overrightarrow{T}}\right)}$, because of the definition of $\mathcal{W}^d_{\left(\tilde{\overrightarrow{\Omega}},\tilde{\overrightarrow{T}}\right)}$. In other words, a static point and a dynamic one can not be discriminated if the motion of the later differs from the ego-motion of a quantity smaller than the extraction noise.

Yet, the presence or absence of a correspondent to m in $\mathcal{W}^m_{\left(\tilde{\overrightarrow{\Omega}},\tilde{\overrightarrow{T}}\right)}$, only allows to distinguish between static and dynamic points.

B. Independent Motion Detection

In order to refine a subsequent image segmentation, the 3D interval of the disparity space, at $t + \partial t$ is defined :

$$\begin{aligned} \mathcal{R}^m_{\left(\tilde{\overrightarrow{\Omega}},\tilde{\overrightarrow{T}}\right)} &= [x+\mu-\kappa\Delta\mu; x+\mu+\kappa\Delta\mu] \\ &\times [y+\nu-\kappa\Delta\nu; y+\nu+\kappa\Delta\nu] \\ &\times [\delta+\xi-\kappa\Delta\xi; \delta+\xi+\kappa\Delta\xi] \end{aligned}$$

The correspondent of m will be searched in $\mathcal{R}^m_{\left(\tilde{\overrightarrow{\Omega}},\tilde{\overrightarrow{T}}\right)}$. That way, one will be able to differentiate between static and dynamic objects, but also between two different dynamic objects.

This correspondence search is performed by calculating the Zero-mean Sum of Absolute Differences (ZSAD) over a small neighborhood, between m and $c\ \forall\ c \in \mathcal{R}^m_{\left(\overrightarrow{\Omega},\overrightarrow{T}\right)}$.

However if a basic correlation is well adapted to stereo pairing problems (due to the epipolar constraint), in a more general case, periodic (or continuous ones) objects can provide a single point with multiple good correspondence candidates. In order to avoid possible ambiguities, a standard *Winner Takes All* (WTA) approach is not used. Instead, every candidate c that has a ZSAD score satisfying the following conditions is considered:

$$\begin{cases} ZSAD_c < Tr \\ \frac{ZSAD_c - min\left(ZSAD_{c'}|c' \in \mathcal{R}^m_{\left(\overrightarrow{\Omega},\overrightarrow{T}\right)}\right)}{ZSAD_c} < \alpha \end{cases}$$

Where Tr is a set threshold, usually set around $5 \times Nb$, where Nb is the number of pixels in the neighborhood used to calculate the ZSAD score, and α is an arbitrary tolerance threshold, usually set around 0.1 to 0.3.

The candidate m'_b that is the closest to $m + \Pi_{\left(\vec{\tilde{\Omega}},\vec{\tilde{T}}\right)}(m)$ among the valid ones is retained. If there is no candidate satisfying the previously mentioned conditions, the point m is flagged with a special value as such a situation can mean several things:

- m lies within an occlusion region
- m is the image of a world point that presents a motion larger than the one imaged by $\mathcal{R}^m_{\left(\vec{\tilde{\Omega}},\vec{\tilde{T}}\right)}$
- the disparity at point m is false.

Through this correlation process, every point m in the disparity space is associated with a vector:

$$\overrightarrow{A} = m'_b - m - \Pi_{\left(\vec{\tilde{\Omega}},\vec{\tilde{T}}\right)}(m) = \begin{cases} \mu_A \\ \nu_A \\ \xi_A \end{cases}$$

Representations of this vector field can be found as Fig. 7 and Fig. 9 (see section V for a description of the used color encoding).

C. Mobile Objects Segmentation

Due to the semi-dense nature of the used disparity maps, one target will often be split into several blobs. A hierarchical clustering process [24] is used to group different parts of an object together. The distance used in order to complete such a process is based on motion and disparity. It can be expressed as :

$$\begin{gathered} Dist_{clustering}(a,b) = \left|\arg\left(\overrightarrow{A}(a)\right) - \arg\left(\overrightarrow{A}(b)\right)\right| \\ +W_1\sqrt{\left|\left|\overrightarrow{A}(a)\right|\right|^2 - \left|\left|\overrightarrow{A}(b)\right|\right|^2} \\ +W_2\left|\delta(a) - \delta(b)\right| \\ +W_3 Dist_{Image}(a,b) \end{gathered}$$

Where W_1 ,W_2 and W_3 are empirically determined relative weights between the different components of the distance, arg is the angle between the vector and the horizontal axis and $Dist_{Image}(a,b)$ is the Euclidean distance, in the image space.

Both these pieces of information are needed, to lift some uncertainties. For instance, if a rigid object is independently moving in a given direction, some part of this object can present a low or zero local contrast (*e.g.* some part of the body of a car), so those parts will appear with no independent motion. Disparity information must be used to associate those parts with a wider, independently moving, set.

V. Results & Discussion

For representation purposes, the projection of the $\overrightarrow{A}$ vector field upon the image space (*i.e.* without its disparity component, which is, anyway, the smallest one) will be displayed, using the color encoding illustrated in Fig. 5.

Tests were conduced with 7 stereo sequences from the french LoVE (*Logiciel d'Observations des Vulnerables*) project. The stereo sensor was composed by two identical 640x480 CCD-cameras, equipped with 6 mm lenses, the baseline of this sensor was 58 cm. All those sequences were shot in urban areas and present low to medium traffic.

Fig. 5. Color representation used. Hue and Saturation are determined by the length and angle of the vector. Value is 0 if $m'_b \in \mathcal{W}^m_{(\Omega,T)}$, 0.6 otherwise.

Fig. 8 shows the results of the presented algorithm for the image sequence illustrated in Fig. 6 & 7. The two targets are well defined. Even if the pedestrian on the right hand side is partially occluded, she is well detected, as every visible part of her body is labelled with the same tone. On the other hand, some body parts of the moving car are identified as static. This is due to its constant and untextured nature, and a way to circumvent this has already been exposed.

It is important to note the road post on the right side of the road. One can see in Fig. 7, that its upper part was attributed a wrong disparity value[2]. This disparity error yields to a motion prediction error. One can also notice some smaller errors, due to false correspondences in the correlation process.

Fig. 9 shows the result of our extraction process. Both targets are accurately located. Besides the road post, some false positives are presents. These are due to false correlation. At this time, there is no filtering process going on in order to eliminate such false positives. Figs. 10 & 11 illustrate the independent motion representation and target extraction for another part of the same sequence. In this part of the sequence, both targets (foreground and background pedestrians) are also well extracted. Figs. 12, 13, 14 & 15 present the extraction results, as well as independent motion representation for two other different sequences. Figs. 12 & 13 present a sequence in which the rotational component of the ego-motion is dominant with respect to the translational one. Figs. 14 & 15 present a much more complex scene. The images contain six different targets, with different motions.

VI. Conclusions & Future Works

A. Conclusions

A stereo-motion based algorithm that works on two consecutive disparity maps[3] and a set of feature correspondences

[2]this is due to the periodic nature of the post pattern.

[3]The frequency of acquisition of those disparity map can range from 2 Hz to 25 Hz, in other ways, some detection can still be achieved with highly time-separated events.

Fig. 6. Image extracted from a real test sequence. The vehicle within the red bounding box is driving forward and initiating a left bend. Within the green box lies the head of a partially occluded pedestrian walking from the right hand side of the field, toward the left-hand side.

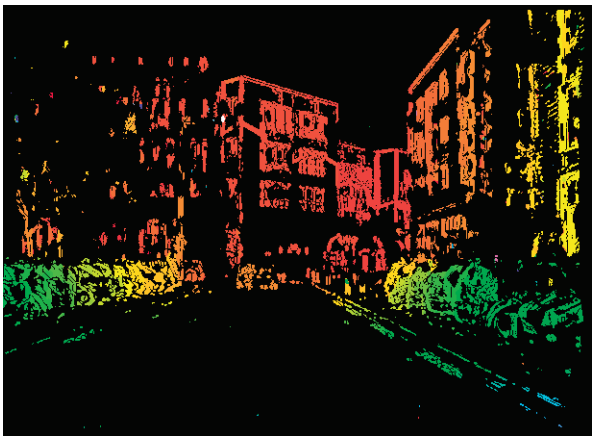


Fig. 7. Disparity Map computed with the method described in [10], from the image pair illustrated by Fig. 6. If one takes a close look, one will notice that a post on the right side of the road presents a disparity error. This is due to the periodic nature of the post pattern, *i.e.*, the upper part of the post is matched with the wrong post in the left image.

is presented here. First the ego-motion of the stereo-rig is estimated, in order to synthesize the corresponding displacement field in the disparity-space (*i.e.* optical flow extended to disparity variations). The precision of this displacement field evaluation for every point in the disparity space is estimated through error propagation. Through a robust correlation method, the points that don't fit in the displacement field can be detected. The work presented here uses this detection in order to achieve localization of independently moving objects. Though, it's worth noting that those points can also belong to disparity map errors.

B. Future Works

As seen previously, the presented method allows the detection of image-based regions that don't validate a majority motion model. These regions can be either an independent moving object, or a disparity error. Thus, this can be used to obtain a correction of the disparity map. Instead of the WTA approach used, several stereo-candidates can be stored for every point. That way, if a point does not comply with the motion estimation for a certain disparity value, but complies with it for another likely disparity-value, disparity can be re-estimated, assuming that the point is static. Errors like the one illustrated in Fig. 7 could be avoided. Such an algorithm is currently being studied.

Besides, the first drawback of this method is its computational cost. As a matter of example, on a standard laptop, a non-optimized C++ implementation runs at 5 frames (640x480) per second. However, all the involved processes present high data parallelism. So, besides using algorithmic means (*e.g.* pyramidal approach), the use of SIMD-optimized coding, or massively parallel computer architectures like GPGPU, or the CELL processor are considered. All those possibilities are currently studied.

Moreover, the method presented here relies only upon the analysis of two consecutive frames. Using some integration over time could improve the elimination of false detections and the identification of small motions. It could be realized by tracking all supposed obstacles through time and disparity space, for instance with a Kalman filter or a mean-shift approach. Such a tracking could also allow to circumvent the dominant motion hypothesis. For instance, if a large moving vehicle occupies a major part of the camera field, its motion will be interpreted as the ego-motion. Tracking this object would allow to label certain parts of the image as dynamic, feature points from within these parts would no longer be used for the ego-motion extraction. Moreover, ego-motion estimation can also be improved through the use of time integration. For instance, a Kalman filter can be used to predict ego-motion, in order to avoid potential aberrations. Such an aberration could be due to the sudden arrival of a dominant moving object in the camera field or to a low-contrast landscape, which can lead to a reduced number of extracted feature points.

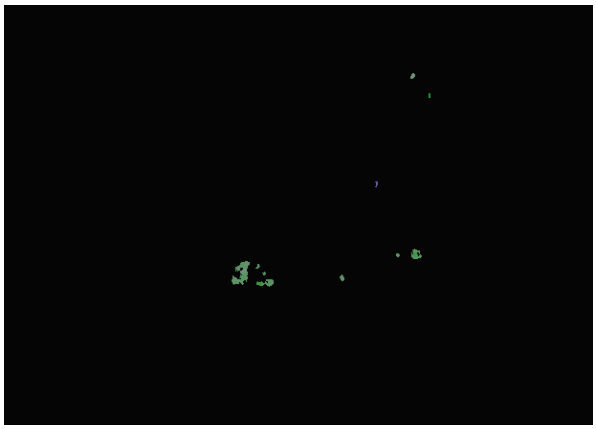


Fig. 8. The car and the pedestrian's head are well defined. Please note that a median blur is applied to the resulting image, in order to eliminate some noise points.

Fig. 9. Final Segmentation Realized. One can note the extraction of the road post, as well as two small false positives. Those false positives are due to false correlations. The targets, however are well defined.

Fig. 10. Image extracted from the same sequence as Fig. 6. The two pedestrians are well extracted, even the one in the background.

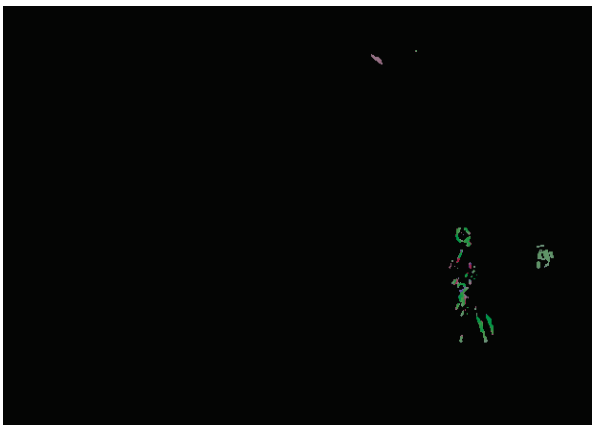


Fig. 11. Independent motion representation. One can note the pedestrian in the background.

VII. Acknowledgements

The authors would like to thanks Dr David Demirdjian for his precious help regarding the extraction of the ego-motion, as well as for his support. The authors would also like to thank Digiteo for its support to their projects.

Fig. 12. This image is extracted from a different sequence, the stereo sensor is moving forward and turning to the left. The white car is well defined and detected. On the other hand the pedestrians on the right are standing still. A stereo-only detection algorithm would have detected them as potential obstacles in spite of they are on the sidewalk.

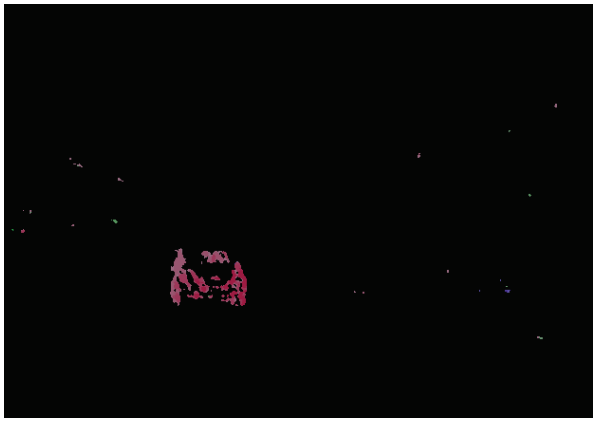


Fig. 13. Independent motion representation relative to the image in Fig. 12. One can note that the front of the vehicle appears with a wider (*i.e.* a more saturated tone) movement than its back. This is consistent with the motion of the car.

References

[1] A.D. Milner and M.A. Goodale, *The Visual Brain in Action* in Oxford University Press, 1996

[2] E. Dickmanns, *The Development of machine vision for road vehicles in the last decade*, IEEE Intelligent Vehicles Symposium, pp 268-281, vol.1, 2002

[3] M.Irani, B. Rousso and S. Peleg, *Recovery of Ego-Motion Using Region Alignment*, IEEE Transactions on Pattern Analysis and Machine Vision, pp 268-272, vol 19, n3, March 1997

[4] Y. Dumortier, I. Herlin and A. Ducrot, *4D-Tensor Voting Motion Segmentation For Obstacle Detection in Autonomous Guided Vehicle*, IEEE Intelligent Vehicles Symposium, pp 379-384, june 2008

[5] S. Beauchemin and J. Barron, *The Computation of Optical Flow*, Association for Computing Machinery Computing Surveys, pp 433-466, vol. 27, issue 3, september 1995

Fig. 14. This scene is more complex than the previous ones, as there are many targets, moving in different directions. However, our method extract them all. The vehicles in the background aren't detected because a minimum disparity under which data is not processed is set. This was done in this sequence only, because of the many moving vehicles in the background.

Fig. 15. Independent motion representation relative to Fig. 14

[6] G. Stein, O. Mano and A. Shashua, *A Robust Method for Computing Vehicle Ego-Motion*, IEEE Intelligent Vehicles Symposium, pp 362-368, , 2000

[7] B. Horn and B. Schunck, *Determining Optical Flow*, Artificial Intelligence, pp 185-203, vol. 17, 1981

[8] B. Lucas and T. Kanade, *An Iterative Image Registration Technique With an Application to Stereo-Vision*, Image Understanding Workshop, pp 121-130, 1981

[9] T.A. Williamson, *A High Performance Stereo Vision System For Obstacle Detection*, PhD Dissertation, Robotics Institute Carnegie Mellon University, Pittsburg, 1998

[10] N. Hautiere, R. Labayrade, M. Perrollaz and D. Aubert, *Road Scene Analysis by Stereovision : a Robust and Quasi-Dense Approach*, International Conference on Control, Autmation, Robotics and Vision, pp 1-6, 2006

[11] S. Heinrich, *Fast Obstacle Detection Using Flow/Depth Constraint*, IEEE Intelligent Vehicles Symposium, pp 658-665, vol. 2, June 2002

[12] U. Franke, C. Rabe, H. Badino and S. Gehrig, *6D-Vision : Fusion of Stereo And Motion For Robust Environment Perception*, Lecture Notes in Computer Science - Pattern Recognition, pp 216-223, vol. 3663, 2005

[13] D. Demirdjian and T. Darrell, *Motion Estimation From Disparity Images*, IEEE International Conference on Computer Vision, pp 213-218, vol. 1, July 2001

[14] A. Howard, *Real-Time Stereo Visual Odometry for Autonomous Vehicles*, IEEE Intelligent Robots and Systems, pp 3946-3952, September 2008

[15] H. Badino, *A Robust Approach For Ego-Motion Estimation Using A Mobile Stereo Platform*, Lecture Notes in Computer Science - Complex Motion, pp 198-208, vol. 3417, 2007

[16] A. Taludker and L. Matthies, *Real-Time Detection of Moving Objects from Moving Vehicles Using Dense Stereo and Optical Flow*, IEEE International Conference on Intelligent Robots and Systems, pp 315-320, Sendai, Japan, Sept. 2004

[17] D. Scharstein and R. Szeliski, *A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms*, International Journal of Computer Vision, 47 (1/2/3):7-42, April-June 2002

[18] C. Harris and M. Stephens, *A combined Corner and Edge Detector*, Alvey Vision Conference, pp 147-151, 1988

[19] H. Bay, A. Ess, T. Tuytelaars and L. Van Gool, *SURF : Speeded-Up Robust Features*, Computer Vision and Image Understanding, pp 346-359, vol. 110, n3, 2008

[20] N. Suvonvorn, S. Bouchafa and B. Zavidovique, *Marrying level lines for stereo or motion*, Internationational Conference on Image Analysis and Recognition, Toronto, Canada, sept 28-30 2005, Proceedings, Lecture Notes in Computer Science 3656 Springer 2005.

[21] M.A. Fischler and R.C. Bolles, *Random Sample Consensus : A Paradigm For Model Fitting with Applications to Image Analysis and Automated Cartography*, Communication of the Association for Computing Machinery, pp 381-395, vol 24, june 1981

[22] R. Hartley and A. Zisserman *Multiple View Geometry in Computer Vision*, Cambridge University Press, 2003

[23] D. Gruyer, C. Royere, N. du Lac, G. Michel and J-M. Blosseville, *SiVIC and RTMaps, interconnected platforms for the conception and the evaluation of driving assistance systems*, IEEE Conference on Intelligent Transportation Systems, 2006

[24] S.C. Johnson, *Hierarchical Clustering Schemes*, Psychometrika, pp 241-254, vol. 32, n 3, 1967

Available online at www.sciencedirect.com
SCIENCE DIRECT
Image and Vision Computing 24 (2006) 70–79
image AND vision COMPUTING
www.elsevier.com/locate/imavis

Efficient cumulative matching for image registration

Samia Bouchafa *, Bertrand Zavidovique

University Paris XI, Institut d'Electronique Fondamentale 91405 Orsay Cedex, France

Received 9 February 2004; received in revised form 16 September 2005; accepted 24 September 2005

Abstract

A new level-line registration technique is proposed for image transform estimation. This approach is robust towards contrast changes, does not require any estimate of the unknown transformation between images and tackles very challenging situations that usually lead to pairing ambiguities, like repetitive patterns in the images. The registration by itself is performed through efficient level-line cumulative matching based on a multi-stage primitive election procedure. Each stage provides a coarse estimate of the transformation that the next stage gets to refine. Even if we deal in this paper with similarity transform (rotation, scale and translation), our approach can be adapted to more general transformations.

Keywords: Registration; Level-lines; Primitive matching

1. Introduction

Registration methods are used to match two or more images taken from different viewpoints, from different sensors or at different times. These last years, various applications including stereovision, 2D/3D motion analysis, pattern recognition or image/video mosaicing, need to register images at some stage. The increased number of approaches motivates many good quality surveys, taking several criteria into account [4,17]. Among other classification criteria, one can quote:

- The nature of the work domain: frequencial or spatial. Frequency approaches are based oil phase correlation and exploit basic properties of the Fourier (or wavelet) transform [24]. Spatial approaches are based on spatial primitive extraction.
- The nature of image transformation (euclidian, similarity, affine, projective or elastic) that implies given types of invariants (length, angle, length ratio, incidence, cross ratio, etc.).
- The chosen primitives. Most frequently used ones are particular points [14,20], edges [21], contours [18], surfaces [23], regions [10], lines [27], or Fourier descriptors [15].
- The similarity measure that depends again on the selected primitives since it measures some similarity between them. Examples of basic similarity measures are: cross-correlation [25], sum of absolute differences [3], or Leven-stein's distance [12].
- The search space and the matching strategy that relate to computation savings. In short, strategies are based on explicit primitive match (e.g. exhaustive search, dynamic programming (12), relaxation labeling [26], Hough transform [2]), or implicit one (e.g. quadratic error minimization [28], linear programming [1]).

Our study stems from three basic observations: first, images to register were seldom shot in even lighting conditions. Yet, commonly used features remain sensitive to contrast changes. Either they depend on gray-level values (points, regions), or they suffer a lack of contrast-independent extraction methods (edges, contours) [6]. Second, quite common ambiguities, for instance in robotics pattern periodicities, are seldom tackled explicitly by matching strategies, although they ultimately hamper a matching process. Third, system's robustness is likely to benefit from progressive decision, so that a guess on transform parameters be available quickly for control, then to be refined if conditions allow. Our approach does specifically address such problems. We select an adapted feature, that is level-line. Level-lines extracted from images are robust to contrast changes [6,19]. Other variables could do: Li et al. [16] for example exploit chromaticity, i.e. independent from intensity. But, we used level-lines successfully in the past for motion analysis [5,11], in particular for outdoor scenes where the luminosity invariance assumption is no longer true. Moreover, they are particularly adapted to a pairing strategy based on voting. Their

* Corresponding author. Tel.: +33 1 69 15 40 07.
E-mail address: bouchafa@ief.u-psud.fr (S. Bouchafa).

0262-8856/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.imavis.2005.09.013

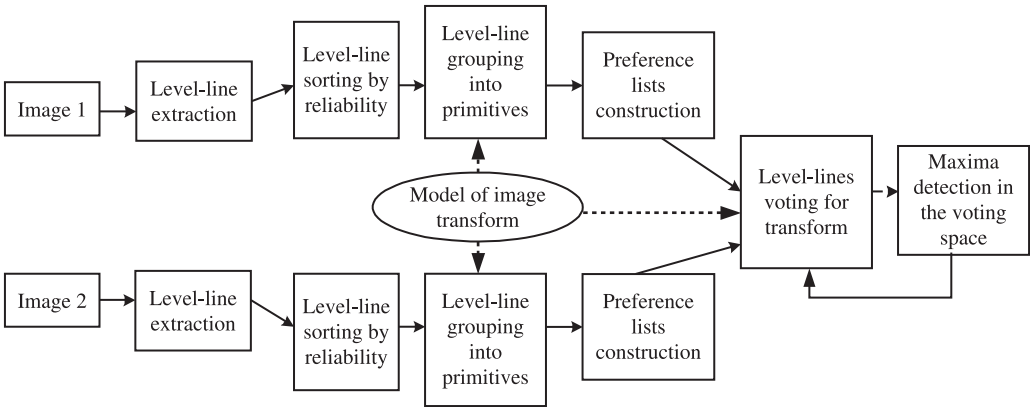


Fig. 1. General overview of our approach.

significant number per image involves strong local redundancies and makes it possible to consider a cumulative mapping process, where each couple of primitives votes in turn according to some reliability. To down-size the voting space, we exploit a decision technique adapted to bipartite graphs known as "stable marriages" [13]. Each primitive creates a preference list of primitives in the other image, sorted from more to less similar according to adapted metrics. These preferences balance the decision-making process that proceeds by several steps: most reliable couples vote first, others are requested then to progressively confirm or contradict the initial vote. The selected transformation collects the maximum votes. That way, in case of ambiguity, i.e. several comparable local maxima in the voting space, a small detail can take it all.

Our paper is organized as follows: first, the level-line extraction process is described. It provides a set of lines with features. Second, level-lines are sorted according to their reliability and paired to form a primitive. Then, the preference-list construction is explained. Eventually, the image transform estimation, thanks to a multi-pass voting process, is outlined in more details (see Fig. 1).

2. Level-lines extraction

Let $I(\mathbf{p})$ denote the intensity of the image at the pixel location $\mathbf{p}(x, y)$. The level set λ of I is the set of pixels $\mathbf{p}$ with an intensity greater than λ.

$$\aleph_\lambda^I = \{\mathbf{p}/I(\mathbf{p}) \geq \lambda\}$$

The level-line associated to the level set $\aleph_\lambda^I$ is its border. One could extract all level sets from the image by using a series of thresholds. Our approach is less time-consuming and more effective: we extract groups of level-lines belonging to the same range of level sets. In fact, we exploit an elementary property of level sets: they are included one in another. That is: $\forall \lambda > \mu \aleph_\lambda^I \subset \aleph_\mu^I$. Thus, level-lines could be locally juxtaposed but they never could cross. Therefore, we propose a simple recursive extraction process that tracks groups of level-lines (the superimposed ones) until they separate. Along the search, subparts are isolated where lines are shown to be straight. This process never considers absolute gray-level values but only the relative order between them. It starts at each point $\mathbf{p}_o(x, y)$, determines which ones among its four neighbours $\mathbf{p}_o(x, y+1)$, $\mathbf{p}_o(x-1, y)$, $\mathbf{p}_o(x, y-1)$ or $\mathbf{p}_o(x+1, y)$ are successors, initiating at most four possible paths (see Fig. 2). Each selected successor becomes the current point $\mathbf{p}_k$ and the process repeats until stopping criteria get false.

$\mathbf{p}_{i=1,4}^k$ is a successor of $\mathbf{p}_k$, at step k if all following conditions are met:

Condition 1 At least one level-line passes between q and r. That is: $|I(r_k)'I(q_k)| \geq$ threshold.

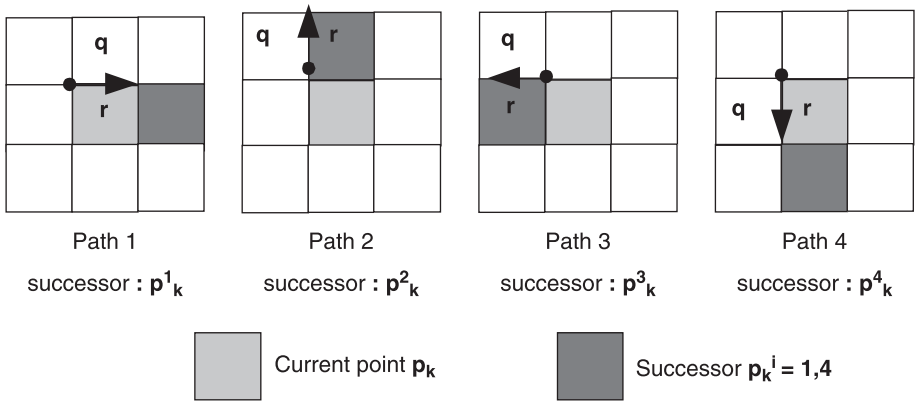


Fig. 2. The four possible paths starting from a point $\mathbf{p}$, at step k of the recursion, and its four possible successors associated with each path. $\mathbf{q}$ and $\mathbf{r}$ represent respective pixels on both sides of the chosen path.

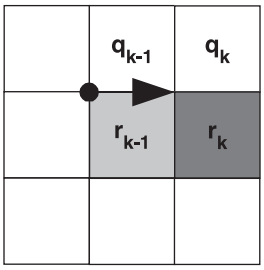


Fig. 3. One has to assert tracking the same group of lines, corresponding to the same range of level sets.

In order to take quantification effects into account, the threshold is set equal to 2 (quantification step + 1). It means that we extract all level-lines without any selection at this stage. Let us underline that our primitive extraction process could need no threshold.

Condition 2 The tracked level-line associated to the chosen path belongs to the same group of level-lines being tracked from the beginning: (Fig. 3)

$$[\min(I(\mathbf{q}_{k-1}), I(\mathbf{r}_{k-1})), \max(I(\mathbf{q}_{k-1}), I(\mathbf{r}_{k-1}))]$$

$$\cap\, [\min(I(\mathbf{q}_k), I(\mathbf{r}_k)), \max(I(\mathbf{q}_k), I(\mathbf{r}_k))] \neq \varnothing$$

Condition 3 The interior (vs. exterior) of the corresponding level sets—associated to the tracked level-lines— is kept on the same side.

That is: $I(\mathbf{r}_k)$ is always greater or always smaller than $I(\mathbf{q}_k)$ for the same group of tracked level-lines.

Condition 4 The tracked level-lines are still straight.

This condition is checked in computing the maximum distance from already traversed points to the cord [initial point, current one]. This distance should not exceed a threshold. Of course, the already traversed points are not stored. Only the crack[1] code corresponding to the path is saved at each step, the reference system origin being set to the initial point $\mathbf{p}_o$ in order to calculate the line equation. (Fig. 4)

This iterative process stops when one of the above conditions is not verified any more. Then it is possible to associate to the starting point $\mathbf{p}_o$(Fig. 5):

(1) the mid-point $\mathbf{p} = (\mathbf{p}_o + \mathbf{p}_k/2)$
(2) the approximate orientation of the straight tracked group of level-lines $\theta_{\overrightarrow{\mathbf{p}_o\mathbf{p}_k}}$
(3) the mean contrast across the tracked level-lines: $c = \left(\sum |I(\mathbf{r}_k) - I(\mathbf{q}_k)|/k\right)$
(4) the length of the tracked level-lines: $l = ||\overrightarrow{\mathbf{p}_o\mathbf{p}_k}||$

Using this information, it is easy to group all points belonging to the same lines. Results are reorganized no longer in terms of starting points but in terms of lines and their characteristics.

Definition 5. A "line" is a group of superimposed straight level-lines obtained from the tracking process.

[1] The crack code corresponds here to the 4-connexity freeman code.

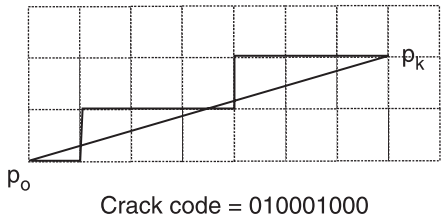


Fig. 4. In order to determine if the tracked level-lines fit a line, we just compute the distance $d_j = |ax_j + by_j + g|$ between each traversed point $\mathbf{p}_j$ (x_j, y_j) and the line $\overrightarrow{\mathbf{p}_o\mathbf{p}_k}$ of equation $ax + by + g = 0$ where $a^2 + b^2 = 1$.

Indeed, the level-line extraction procedure exhibited all level-lines, but actually they came out as groups of straight segments belonging to the same range of level sets (see condition 2). The end result of the level-line extraction process is then the set of all straight pieces of level-lines $\{L_I^i\}_{i=1,\eta}$ belonging to image I with associated features:

$$\overrightarrow{L_I^i} = [\mathrm{p}\theta lc]^T$$

3. Lines sorting

Level-lines do not have all the same visual importance and do not always coincide with perceptual contours. Some of them separate very close level sets in term of gray values. This is analog to very low edge slopes when the contrast between regions is light. These level-lines have stronger probability to disappear under natural contrast changes. Moreover, other level-lines are very short, meaning that they are associated to very small level sets likely to be noise generated. There are several ways of selecting lines, which correspond the best to perceived contours. Among them, let us quote the work by J. Froment [8,9] or N. Paragios & R. Deriche [22]. We choose here to take advantage from the perceptual differences between extracted lines. We then define reliability from line length and contrast. Less reliable lines are not completely discarded from the decision process: they contribute with a weaker credit. Lines are classified according to mean and standard deviation of lengths and contrasts over the image, $\bar{l}$, σ_l and $\bar{c}$, σ_c, respectively. We classify lines into three categories. Most important level-lines correspond to those with higher contrast and length than $\bar{l} + \sigma_l$ and $\bar{c}$, σ_c, respectively. The second category is assigned to

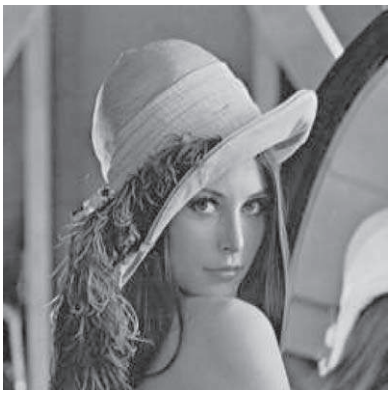


Fig. 5. Original image.

those having large enough length too but lower contrast ($l > \bar{l} + \sigma_l$ and $\bar{c} < c < \bar{c} + \sigma_c$) and the last one gathers the shortest yet with high enough contrast ($\bar{l} < l < \bar{l} + \sigma_l$ and $c > \bar{c} + \sigma_c$). The remaining level-lines are rejected (see Fig. 10). Of course thresholds can be adapted to applications (Figs. 6–10).

4. Lines grouping onto primitives

Lines grouping depends on the nature of the transform we would like to estimate (parameters number). Indeed, each line will be abstracted into just one point, the mid-point (thus: two coordinates). Then, to deal with four parameters transform (dx, dy, α, ψ)—dx and dy are the translations along x and y, respectively, ψ the uniform scale and α the planar rotation around z axis—we decide to consider a couple of lines as one primitive for matching. Line's length and orientation are no longer invariant under similarity transform. Therefore, we have to define invariant features under the considered transformation as primitive. We pick "ratio of lengths" and "difference of orientation" between any two lines.

Definition 6. A primitive is a set of p lines.

Definition 7. An "angle" is a set of $p=2$ lines.

Definition 8. A "couple" of primitives **u** and **v** is composed of two primitives extracted from two images.

Let us consider two given lines $\overrightarrow{L_I^i}$ and $\overrightarrow{L_I^j}$ with their associated characteristics:

$$\overrightarrow{L_I^i} = [\mathbf{p}_i \theta_i l_i c_i]^T \quad \text{and} \quad \overrightarrow{L_I^j} = [\mathbf{p}_j \theta_j l_j c_j]^T$$

Then, the couple formed by these lines is:

$$\overrightarrow{C_I^{ij}} = [\mathbf{p}_i \mathbf{p}_j \Delta\theta \wr c_i c_j]^T \quad \text{with} \quad \Delta\theta = \theta_i - \theta_j, \quad \wr? = \frac{l_i}{l_j}$$

This couple is a valid one if and only if two conditions are verified:

(1) Considered lines are likely enough not-collinear: $\varsigma\theta > \varsigma\theta_{\min}$, ($\varsigma\theta_{\min}$ to be set in conformity with the maximum curvature in the extraction process)
(2) Considered lines are close enough in terms of distance: $||\overrightarrow{\mathbf{p}_i\mathbf{p}_j}|| < \text{dist}_{\max}$ The choice of $\varsigma\theta_{\min}$ and $\text{dist}_{\max}$ depends on the considered transformation in order to avoid matching lines that could disappear.

Fig. 6. In black: all points where at least one level-line passes through.

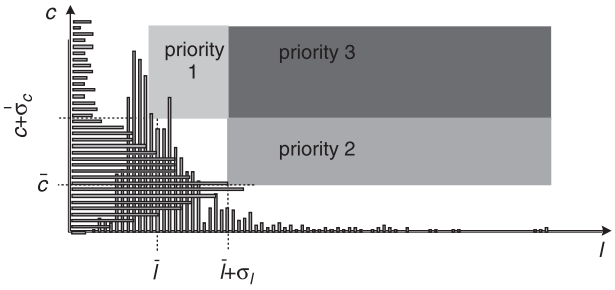


Fig. 7. Primitive classification according to lengths and mean contrast with respect to distributions.

Couples are formed by considering first the top of the line list (most reliable), so that the created list of couples will be also sorted by decreasing line reliability. Since planar rotations are dealt with here, the algorithm to sort couples is made systematic (see the example below). For more complex transforms or more ambiguous and noisy scenes, parameters as h or ($p_i q_{ij}$, $p_j q_{ij}$) (see Fig. 11) or the continuity, from lines i to j, between mean gray levels on the right ($\mu_{ir} \rightarrow \mu_{jr}$) and the left side ($\mu_{il}, \rightarrow \mu_{jl}$) of the line, etc. should be considered to make main voters a priori even more dependable (Fig. 11).

Example 9 Suppose we have a list of 6 lines sorted by reliability. The first couple is (0,1), then (0,2), (0,3), (1,2), (0,4), (1,3), (0,5), (1,4), (2,3), see Fig. 12.

5. Primitive matching

The primitive sorting led to consider a multi-stage matching process where each phase involves a new category of primitives to vote according to its reliability (position in the sorted list of primitives). That aims to improve the decision dynamics in shortening greedy procedures as voting. Indeed, the initial cross-selection of primitives between images is a simplified, yet quite combinatorial "stable marriage process" [29]. Prior to any election, each primitive is asked to construct a $\kappa_{\max}$-nearest primitive preference list. The voting process then consists of selecting a potential match according to candidate positions in this list. Let us further detail this procedure.

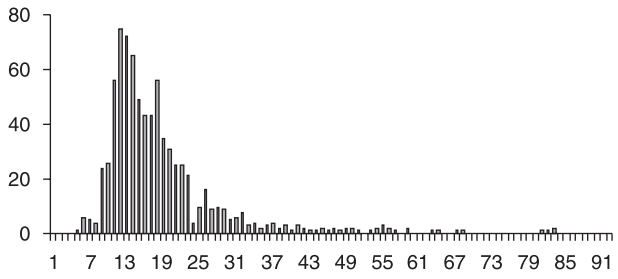


Fig. 8. Example of line length repartition for the above image.

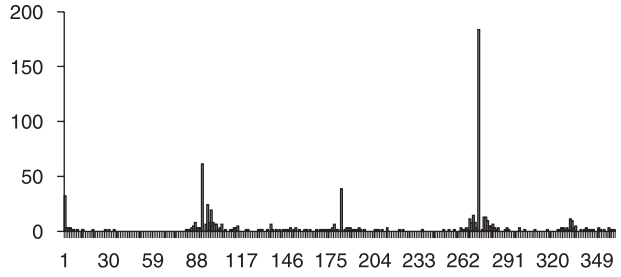


Fig. 9. Example of level-line orientation (between 0 and 359°) repartition for the above image.

5.1. Construction of variable length preference lists

Each primitive $\mathbf{u} \in \{\overrightarrow{C_I^k}\}_{k=1,N}$ in the first image I builds a preference list of primitives sorted among those in image J, from most to least similar one, using a distance measure based on ratio of lengths, orientation differences, and the mean contrast on both sides of the lines. The distance dist between $\mathbf{u}$ and any $\mathbf{v} \in \{\overrightarrow{C_J^m}\}_{m=1,M}$ is simply defined as:

$$\mathrm{dist}(\mathbf{u},\mathbf{v}) = k_1|\Delta\theta_{\mathbf{u}} - \Delta\theta_{\mathbf{v}}| + k_2|\ell_{\mathbf{u}} - \ell_{\mathbf{v}}| + k_3(|c_{i\mathbf{u}} - c_{i\mathbf{v}}| + |c_{j\mathbf{u}} - c_{j\mathbf{v}}|).$$

k_1, k_2, and k_3 allow to adjust the relative importance of each characteristic.

Note that $\sum k_i = 1$ and $k_3 < k_{i=1,2}$ because contrast is not an invariant characteristic that much, yet to be considered with a smaller weight. Of course, each absolute difference is normalized by its maximum possible value in order to get a final distance between 0 and 1 (from more similar to less similar). Each primitive constructs its κ-nearest ($\kappa < \kappa_{\max}$) primitive list using the above distance. It is important to note that the size of the preference list varies from one primitive to the other. New candidates are inserted into the preference list of a given primitive until the computed distance becomes much higher than those of previously selected items. In fact, the choice of variable-length (but limited to $\kappa_{\max}$) preference lists is necessary to avoid arbitrary elimination of a primitive candidate because of a priori fixed list size (Fig. 13).

5.2. Voting process

We propose to use a matching strategy based on a multi-stage voting process where categories of primitive express their opinion to be accounted for according to their reliability. This strategy proves two main advantages. First, each stage provides a coarse estimate of the transform that could be—even it is still rough—used in real-time applications while the next stage is running to refine the estimate. Second, at a given stage, repetitive patterns present in images can lead to pairing ambiguities: the process which consists of inviting less reliable primitives—in a greater number and lesser trust—to vote, gives

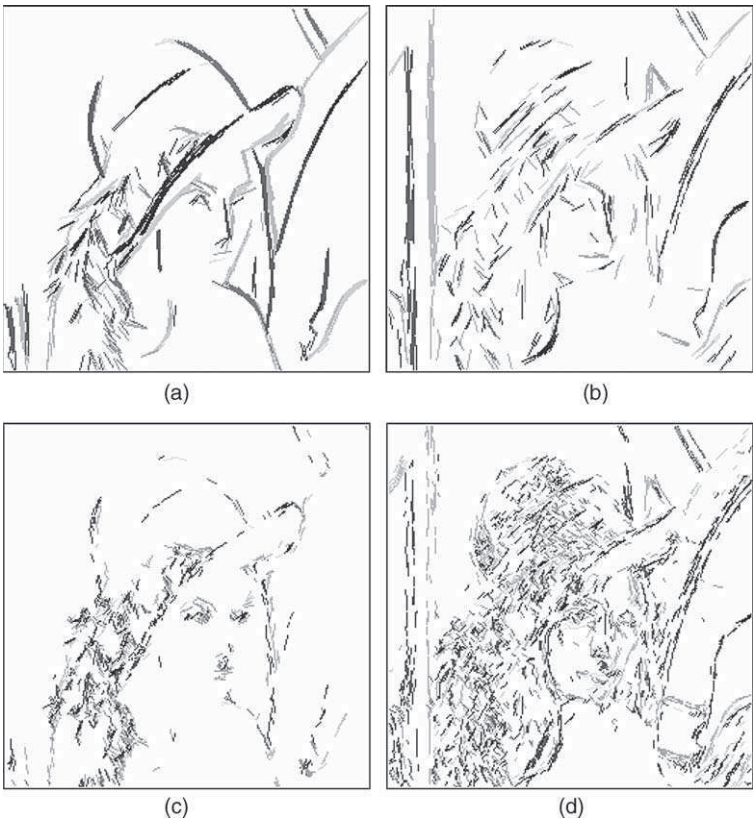


Fig. 10. Results of level-line extraction and sorting. Eight colors are used. For display purpose, each color corresponds to a particular line orientation using a discretization step $\pi/4$. Lines are displayed merely using starting point, ending point and obtained direction. The total number of lines $= \eta$; (a) first $\eta/4$ lines, which corresponds to more reliable; (b) next $\eta/4$; next ($\eta/4$); (d) last ($\eta/4$): less reliable lines.

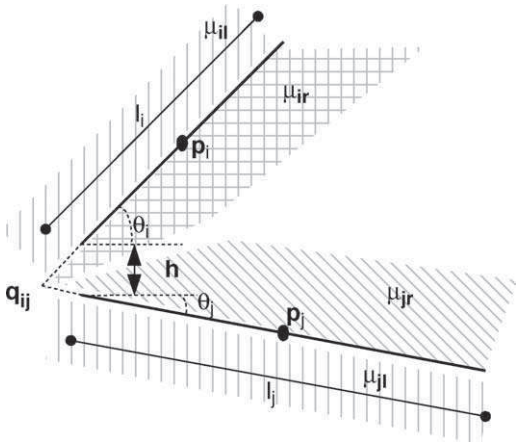


Fig. 11. A couple $\overrightarrow{C_I^{ij}}$ formed by two lines $\overrightarrow{L_I^i}$ and $\overrightarrow{L_I^j}$.

another chance to make the exact solution emerge. The refinement procedure, however, provides an estimate in all cases that focuses the complete decision.

Primitives are classified into $W_{\max}$ categories, where $W_{\max}$ is the chosen number of voting stages. Let us add another characteristic to each couple, its reliability weight w:

$$\overrightarrow{C_I^k} = [\mathbf{p}_i\mathbf{p}_j\Delta\theta\ell c_i c_j w]^T.$$

First, at stage $W_{\max}$, all primitives with high priority weight ($w = W_{\max}$) vote. If a single peak[2] appears in the voting space (a transform obtains absolute majority), then the voting process stops. Else a second stage could begin involving an additional population of voters: the primitives with less priority weight ($w = W_{\max} - 1$). **S** peaks in the voting space are selected according to the configuration of the space (interdistances, heights, extension, …). Peaks are Cumulative values of selected *maxima*, will be enhanced for the next stage by a multiplicative factor **K** in order to privilege reliable primitive decisions. The voting process stops when no more primitive is available or when just one peak does appear collecting absolute majority (one transform collects the maximum votes) (Fig. 14).

The voting space V is a set of three cumulative spaces (**T** for translation, **R** for rotation and **S** for scale). In the translation cumulative space, coordinates (dx, dy) represent shift values and the content **T** (dx, dy) is the global vote in favor of translation (dx, dy) (between $[-\mathrm{d}\mathbf{x}_{\max}, \mathrm{d}\mathbf{x}_{\max}]$ and $[-\mathrm{d}\mathbf{y}_{\max}, \mathrm{d}\mathbf{y}_{\max}]$ respectively). The origin that is identity (translation = (0, 0)) is placed in the middle of the space. In the rotation space, coordinate α represents the rotation angle (between $[\alpha_{\min}, \alpha_{\max}] = [0°, 360°]$). Finally, in the scale cumulative space, coordinate ψ represents the scale factor between $[\psi_{\min}, \psi_{\max}]$.

Each primitive **u** votes for a deduced transformation in association with each candidate **v** in its preference list. The transformation, that is four unknowns dx, dy, α, and ψ, is estimated locally to that couple by simply solving the system of four equations. Each point provides two equations, one for each coordinate x and y:

[2] Peaks are found by determining local maxima in the voting space.

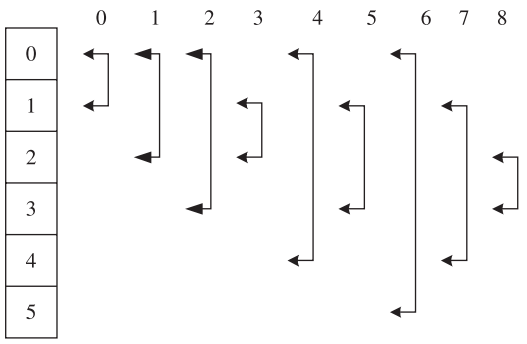


Fig. 12. Primitive list construction from the level-line list. Example with 6 level-lines on the whole. The combinatorics algorithm groups level-lines preserving their reliability order.

$$\mathbf{p}_{k\mathbf{u}}\begin{bmatrix}\psi & 0\\ 0 & \psi\end{bmatrix}\begin{bmatrix}\cos(\alpha) & -\sin(\alpha)\\ \sin(\alpha) & \cos(\alpha)\end{bmatrix}p_{k\mathbf{v}} + \begin{bmatrix}\mathrm{d}x\\ \mathrm{d}y\end{bmatrix}; \quad k = i \text{ or } j$$

In other terms, given:

$$\begin{cases} U = \psi\cos(\alpha) \\ V = \psi\sin(\alpha) \\ Z = \mathrm{d}x \\ T = \mathrm{d}y \end{cases}$$

The previous system rewrites as:

$$\begin{bmatrix} x_{\mathbf{p}_{iv}} \\ y_{\mathbf{p}_{iv}} \\ x_{\mathbf{p}_{jv}} \\ y_{\mathbf{p}_{jv}} \end{bmatrix} = \begin{bmatrix} x_{\mathbf{p}_{iu}} & y_{\mathbf{p}_{iu}} & 1 & 0 \\ y_{\mathbf{p}_{iu}} & x_{\mathbf{p}_{iu}} & 0 & 1 \\ x_{\mathbf{p}_{ju}} & -y_{\mathbf{p}_{ju}} & 1 & 0 \\ y_{\mathbf{p}_{ju}} & x_{\mathbf{p}_{ju}} & 1 & 1 \end{bmatrix}\begin{bmatrix} U \\ V \\ Z \\ T \end{bmatrix} = M\begin{bmatrix} U \\ V \\ Z \\ T \end{bmatrix}$$

then $[UVZT]^T = M^{-1}[x_{\mathbf{p}_{iv}} y_{\mathbf{p}_{iv}} x_{\mathbf{p}_{jv}} y_{\mathbf{p}_{jv}}]^T$ where x and y are midpoints coordinates. Hence:

$$\mathrm{d}x = Z$$

$$\mathrm{d}y = T$$

$$\alpha = arctg\left(\frac{V}{U}\right)$$

Fig. 13. Example of mutual preference.

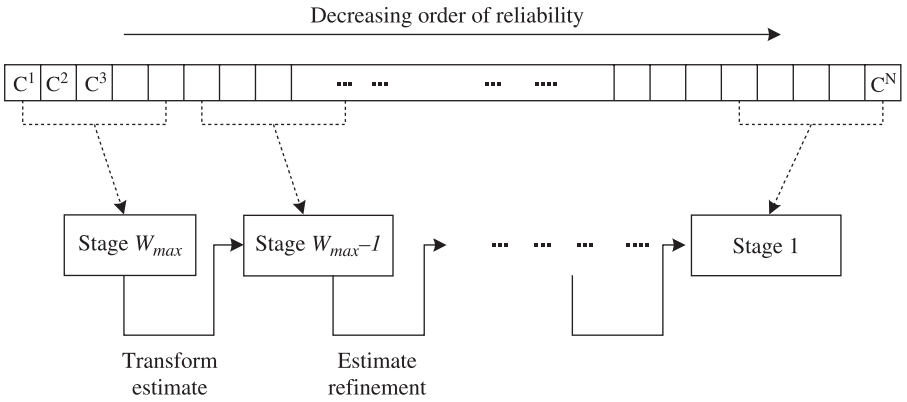


Fig. 14. (a) and (b) Two partial profiles of an aluminum ring. (c) and (d) The voting space represented by an image or a 3D mesh and its maxima which correspond to the different possible shifts. Here, only one pass is used because one significant enough peak appears in the voting cumulative space. (e) The result of our alignment method: shift of (−2,72) which corresponds to the shift peak.

$$\psi = \frac{U}{\cos(\alpha)} = \frac{V}{\sin(\alpha)}$$

Algorithm:
Initialization
{selected maxima} $\{dx^s, dy^s\ \alpha^s, \psi^s\}$
$dx^s = \{-d\mathbf{x}_{\max}, d\mathbf{x}_{\max}\}$,
$dy^s = \{-d\mathbf{y}_{\max}, d\mathbf{y}_{\max}\}$,
$\alpha^s = \{\boldsymbol{\alpha}_{\min}, \boldsymbol{\alpha}_{\max}\}$,
$\psi^s = \{\boldsymbol{\psi}_{\min}, \boldsymbol{\psi}_{\max}\}$.
FOR $t = W_{\max}$, DOWNTO 1 DO
$\forall (dx, dy, \alpha, \psi)\ \mathbf{V}_t\ [dx, dy, \alpha, \psi] = 0$
$\mathbf{S} = t$
FOR each primitive $\mathbf{u} \in [\overrightarrow{C_I^k}]_{k=1,N}$ the image I with $\mathbf{u}_w < t$ DO
FOR $pos = 1, \boldsymbol{\kappa}$ DO
Let $\mathbf{v}_{pos} = (_{\mathrm{u}}[pos]$
$(dx, dy, \alpha, \psi) =$ system solve for $(\mathbf{p}_{i\mathbf{u}}, \mathbf{p}_{j\mathbf{u}}, \mathbf{p}_{i\mathbf{v}}, \mathbf{p}_{j\mathbf{v}})$
$\mathbf{V}_t\ [dx, dy, \alpha, \psi] = \mathbf{V}_t\ [dx, dy, \alpha, \psi] + \Delta v$
END
END
$\{dx^s, dy^s, \alpha^s, \psi^s\} = \mathbf{S} - \max(\mathbf{V}_t[dx, dy, \alpha, \psi])$
$\mathbf{V}_t[\{dx^s, dy^s, \alpha^s, \psi^s\}] = \mathbf{V}_t[\{dx^s, dy^s, \alpha^s, \psi\}] \times \mathbf{K}$
$\mathbf{V}_{t+1} = \mathbf{V}_t$
Stop if just one peak appear in the voting space.
END

The contribution significance Δv of the vote for a given transform (dx, dy, α, ψ) between primitives $\mathbf{u}$ and $\mathbf{v}$ is computed taking into account the following considerations. Let $\Delta_1 = a_1 \times a_2 \times a_3 \times a_4$, where parameters a_1, a_2, a_3, a_4 are adjusted as follows

(1) A potential correspondent $\mathbf{v}_{pos}$ will have a higher contribution if it lays in the beginning of the preference list. a_1 is thus inversely proportional to the position pos of the primitive v_{pos}. Let choose:

$$a_1 = \frac{1}{pos}; \quad a_1 = \left\{\frac{1}{\kappa}, \ldots, 1\right\}, \quad pos \in [1, \kappa]_{\kappa < \kappa_{\max}}$$

(2) At a given stage of the voting process, primitives with category less or t ($t < W_{\max}$ is the current iteration of the voting process) are mate with primitives in the second image. For example, in the first pass $t = 3$ and voters must have priority 3; for the second pass $t = 2$ and voters must have priority 2 or 3, etc. However, primitives with higher priority still vote in higher consideration. a_2 is then proportional to the primitive weight. That is:

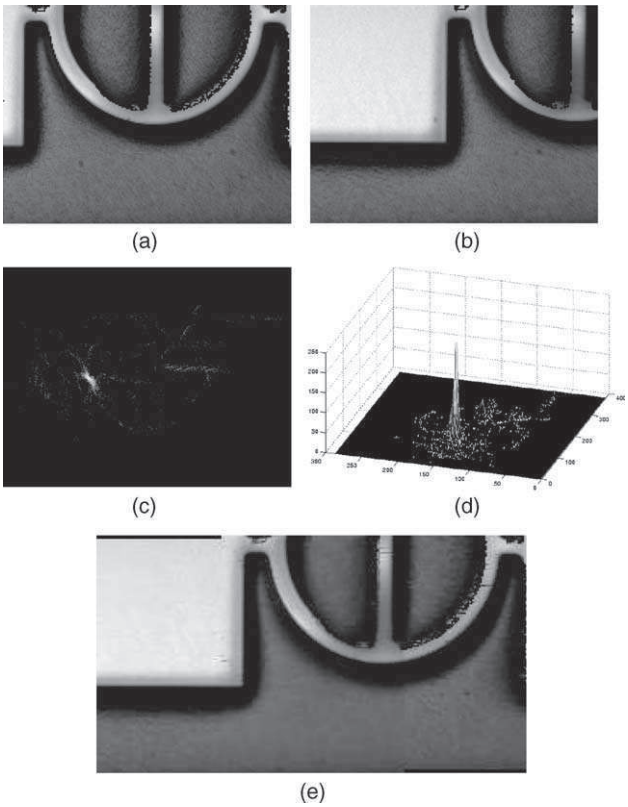


Fig. 15. (a) and (b) Partial profiles of a micro gyrometer electrostatic copper comb. (c)–(e) The voting spaces corresponding to the three vote-iterations. (f) Image warping using the resulting correct transform (shift of −98, 3) in spite of the repetitive patterns. (g) Result obtained by a classical approach.

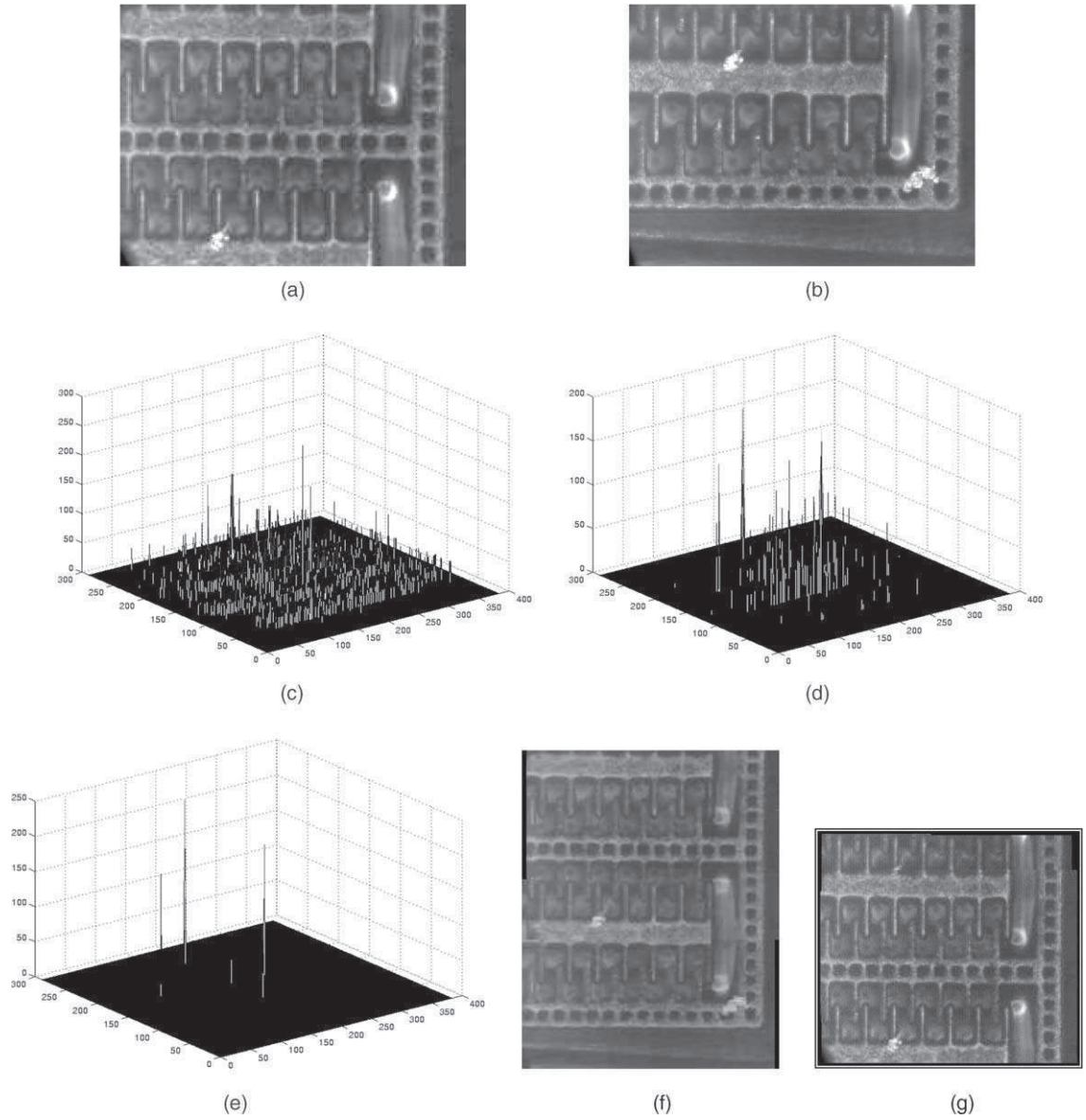


Fig. 16. (a) Image 1. (b) Image 2. (c) Result of the registration process: the gray scale of image 1 is reduced for transparency. The resulting transformation is: $dx = 5$, $dy = 3$, teta = 9, scale = 1, 0096. Registration is satisfactory for control purpose in such applications.

$$a_2 = \frac{1}{w_{\mathbf{u}}}; \quad a_2 = \left\{ \frac{1}{W_{\max}}, \ldots, 1 \right\}$$

(3) Most similar primitives (low distance) must vote with a higher value. Hence the factor:

$$a_3 = (1 - \mathrm{dist}(\mathbf{u}, \mathbf{v})); \quad a_3 \in [0 \ldots 1]$$

(4) Finally, because all a_i where $i = 1, 3$ belong to the range [0, 1], we can use a zoom factor a_4 in order to get integer increments.

6. Results

We have applied our method to register micro-mechanical devices images obtained by an interferometric profilometry technique. This microscopy technique allows to measure the 3D profile of a surface. Resolution is sub-micronic horizontally and nanometric along the vertical axis, likely to further involve sub-pixel refinements. The technique is used for the characterization of μ-mechanical-device behaviors in order to size them up and predict their reliability as electromechanical μ-systems. To extend the field of measurement—limited to a few hundred microns—it becomes necessary to move mechanically the μ-device in front of the lighting and optical system. Then, several partially superimposed 3D profiles are acquired, needing realignment. Due to their 3D nature, the profiles can be represented as pictures and paired thanks to image registration techniques. Aligning profiles then involves a matching process. However, the specificity of our application, due in particular to various imperfections—dust, spots, stains on lenses—of the measurement system, and varied lighting along with the physical shift, makes it difficult to solve using

Fig. 17. Voting space corresponding to Fig. 7 for every parameter: (a) dx, (b) dy, (c) teta, (d) scale.

traditional registration techniques: images do not keep contrast, fixed artifacts do appear and repetitive structures common in such MEMS generate ambiguities against mapping. In addition to this, the implemented method should be fully automated for fundamental-physics users not requiring any arbitrary threshold or a priori knowledge (e.g. minimal overlapping, magnitude or direction of deformation).

We have compared our technique with two classical approaches: phase correlation (using an FFT transform) and intensity distance minimization (using the well-known Levenberg–Marquardt optimization method). None of them give expected results. We have also used a very interesting on-line web demo [7]. Due to the repetitive patterns present in the images, the obtained transform is not correct as can be seen in Fig. 15(g). Following figures give first results of our method in a very challenging situation due to the repetitive patterns, and low contrast (Fig. 16).

7. Conclusion

We have proposed an image registration approach robust to contrast changes that does not require any prior estimate of the unknown shift between images and that takes into account specific problems due to our image acquisition system or to the application, like lighting movements and repetitive patterns. We chose specific primitives with a very suitable property: their contrast change invariance. We then propose a recursive extraction process that never uses directly gray level values but only the relative order between them. The transform extraction and then registration are performed through an efficient matching procedure based on multi-pass voting. It consists of constructing a list of potential candidates for each primitive, the most similar ones according to an adapted metric and based on specific measurements. The winning transformation cumulates the maximum votes. The multi-pass voting is introduced in order to tackle the problem of local maxima—corresponding for instance to the repetitive pattern responses—that could be found in the voting space. Primitives are then invited to vote according to their importance at each pass of the election. Results we got are very encouraging and give expected results in spite of the very challenging situations we have considered.

8. Future work

Our next step is to study the influence of a specific parameter: the length of preference lists. Moreover we would like to draw advantage from the stable marriage technique in order to improve the matching process. Finally, we would like to address other transformations between images. Considering more general transformations leads to group lines by two, three or more depending on the number of transform parameters. The matching process by itself will not change (Fig. 17).

Acknowledgements

We would like to thank Prof. Guna Seetharaman, for fruitful discussions and advice while he was visiting IEF.

References

[1] H.S. Baird, Model-Based Image Matching Using Location, MIT Press, Cambridge, 1985.
[2] D.H. Ballard, Generalizing the hough transform to detect arbitrary shapes, Pattern Recognition Journal 13 (2) (1981) 111–122.
[3] D.I. Barnea, H.F. Silverman, A class of algorithms for fast digital image registration, IEEE Transactions on Computers 21 (2) (1972) 179–186.
[4] L.G. Brown, A survey of image registration techniques, ACM Computing Surveys 24 (4) (1992) 325–376.
[5] S. Bouchafa, Motion detection invariant to contrast changes, Application to detecting abnormal motion in subway corridors, PhD Thesis, University Paris 6, 1998.

[6] V. Caselles, B. Coll, J. Morel, Topographic maps and local contrast change in, natural images, International Journal of Computer Vision 33 (1) (1999) 5–27.
[7] F. Dmitry, C. Kenney, B.S. Manjunath, L.M.G. Fonseca, On Line Registation Demo, University of California, Santa Barbara, 2002. http://nayana.ece.tiesb.edu/registration/
[8] J. Froment, Image compression through level lines and wavelet packets, in: A.A. Petrosian, F.G. Meyer (Eds.), Wavelets in Signal and Image Analysis, Kluwer Academic, Dordecht, 2001.
[9] J. Roment, A compact and multiscale image model based on level sets, Scape Space Theories in Computer Vision, Lecture Notes in Computer Science 1682, Proc. of Sec. Int. Conf. Scale-Space '99, pp 152–163, 1999.
[10] A. Goshtasby, G.C. Stockman, C.V. Page, A region-based approach to digital image registration with sub-pixel accuracy, IEEE Transactions on Geoscience and Remote Sensing 24 (3) (1986) 390–399.
[11] F. Guichard, S. Bouchafa, D. Aubert, A change detector based on level sets, ISMM2000, Palo Alto, June 2000.
[12] Y. Le Guilloux, A matching algorithm for horizontal motion, application to tracking, IEEE Proceeding of the 8th International Conference on Pattern Recognition, Paris, 1986.
[13] D. Gusfield, R.W. Irwing, The Stable Marriage Problem—Structure and Analysis, MIT Press, Cambridge, MA, 1989.
[14] L.N. Kanal, B.A. Lambird, D. Lavine, G.C. Stockman, Digital registration of images from similar and dissimilar sensors, Proceedings of the International Conference on Cybernetics and Society, 1981.
[15] P. Kuhl, C. Giardina, Elliptic fourier features of a closed contour, Computer Graphics and Image Processing 18 (1982) 236–258.
[16] Z.N. Li, Z. Tauber, M.S. Drew, Locale-based object search under illumination change using chromaticity voting and elastic correlation, IEEE Conference on Multimedia and Expo 2000.
[17] J.B.A. Maintz, M.A. Viergever, A survey of medical Image registration, Medical Image Analysis, vol. 2, Oxford University Press, Oxford, 1998.
[18] G.G. Medioni, R. Nevatia, Matching images using linear features, IEEE Transactions on Pattern Analysis and Machine Intelligence 6 (6) (1984) 675–685.
[19] P. Monasse, F. Guichard, Fast computation of a contrast-invariant image representation, IEEE Transactions on Image Processing 9 (5) (2000) 860–872.
[20] H. Moravec, Rover Visual Obstacle Avoidance, IJCAI, Vancouver, 1981, pp. 785–790.
[21] M.L. Nack, Rectification and registration of digital images and the effect of cloud detection, Proceedigns of Machine Processing of Remotely Sensed Data, 1977, pp. 12–23.
[22] N. Paragios, R. Deriche, A PDE-based level set approach for detection and tracking of moving objects, Proceedings of ICCV'98, 1998, pp. 1139–1145.
[23] C.A. Pelizzari, G.T.Y. Chen, D.R. Spelbring, R.R. Weichselbaum, C.T. Chen, Accurate three dimensional registration of CT, PET and/or MR images of the brain, Journal of Computer Assisted Tomography 13 (1989) 20–26.
[24] B.S. Reddy, B.N. Chatterji, An FFT-based technique for translation, rotation and scale-invariant image registration, IEEE Transactions on Image Processing 5 (8) (1996).
[25] A. Rosenfeld, A.C. Kak, Digital Picture Processing, Academic Press, New York, 1982.
[26] L.G. Shapiro, R.M. Haralick, Matching relational structures using discrete relaxation, Syntactic and Structural Pattern Analysis Theory and Applications, World Scientific Pub. Co., Teaneck, NJ, 1990.
[27] G.C. Stockman, S. Kopstein, S. Benett, Matching images to models for registration and object detection via clustering, IEEE Transactions on Pattern Analysis and Machine Intelligence 4 (3) (1982) 229–241.
[28] R. Szeliski, Image mosaicing for tele-reality applications, in: Proceedings of the IEEE Workshop on Applications of Computer Vision, 1994.
[29] K. Zemirli, G. Seetharaman, B. Zavidovique, Stable matching or selective junction points grouping, IEEE JCIS 2000, Atlantic City, NJ, February 2000.

Marrying Level Lines for Stereo or Motion

Nikom Suvonvorn, Samia Bouchafa, and Bertrand Zavidovique

Institut d'Electronique Fondamentale, Université Paris 11,
Bâtiment 220 - 91405 Orsay Cedex

Abstract. Efficient matching methods are crucial in Image Processing. In the present paper we outline a novel algorithm of "stable marriages" that is also fair and globally satisfactory for both populations to be paired. Our applicative examples here being stereo or motion we match primitives based on level lines segments, known for their robustness to contrast changes. They are separately extracted from images, and we draft the corresponding process too. Then for marriages to be organised each primitive needs to be given a preference list sorting potential mates in the antagonist image: parameters of the resemblance founding preferences are explained. Eventually all operators above are embedded within a recursive least squares method and results are shown and compared with a successful Hough based matching that we had used so far.

1 Introduction

Efficient matching methods are needed in all areas of Image Processing, ranging from Segmentation - e.g. motion detection or 3D reconstruction from stereo - to actual Pattern Recognition - e.g. model fitting or classification. Efficiency then gets multiple meanings and can address properties as different as easy data extraction and coding format, model simplicity, limited prior assumptions, robustness against ambiguities, conflict freeness etc. not to forget computability. Within that frame, general enough methods are still to be produced. We tested several, from Dynamic Warping on edge or region chain codes [1] for instance to more recently Hough Transform on level lines [2]. In the latter we showed how using n-tuples of carefully coded level line segments, sorted into several sub populations according to the confidence in them, leads to an efficient multipass voting process. Efficiency here is in the sense of "fighting ambiguities (e.g. repetitive patterns) thanks to a reinforcement of stronger by weaker features at a limited enough computing expense". In a wider approach to segmentation that aimed at exhibiting nD image features in gathering (n-1)D ones [3] − edges from points of interest, regions from edges etc − an interesting paradigm of optimisation on bi-partite graphs was tested against Bayesian techniques and shown to sustain comparison well with them. It is called the "stable marriage problem", of which the main interest is to guarantee no logical contradiction in the pairing process provided two different sets of items are genuinely distinguished to be cross-matched one-to-one and each item sorts every member of the antagonist set in a so-called "preference list". Efficiency in that case stresses disambiguation again and we thought to try the method to limit or skip multipass voting in the

M. Kamel and A. Campilho (Eds.): ICIAR 2005, LNCS 3656, pp. 391–398, 2005.

preceding scheme. The present paper is devoted to preliminary experimentations in that direction for 3D Reconstruction and Motion finding.

It is organized as follows: first, we describe the level-lines' junction extraction resulting into primitives. Second, the preference list construction is explained. Then, the stable marriages algorithm we designed for such matching is outlined. After explaining how outliers are eliminated, we describe shortly the image transformation estimation. Eventually, some results are displayed for comments.

2 Image Features

The feature type plays a significant role in the choice of a matching strategy. Less reliable features may lead to very complex matching processes. The level-line is chosen here for being a robust/reliable feature, its invariance property towards contrast changes is known [4][5]. $I_{\mathbf{p}}$ being the image intensity at pixel $\mathbf{p}$, the *level set* N_λ of image I is made of the pixels which intensities are equal or larger than λ, $N_\lambda = \{\mathbf{p} / I(\mathbf{p}) \geq \lambda\}$. Borders of such level sets are called *level lines* L_λ. One important property of level lines is that they can overlay but cannot cross. Let F_λ be a set of overlaying level lines, called *level line flow*, defined by $F_\lambda = \left\{L^I_\lambda / \lambda \in [u, v]\right\}$. $(v - u)$ is called the *flow extension* $\mathcal{E}$. The method proposed in [6][7] extracts the level lines by tracking such flows . In an image, the point where two level line flows merge or split is called a *flow junction*.

The extraction process, under the form of a recursive automaton, performs in 3 steps. First step consists of finding the level line flows at the left-top of each pixel. The following flows are calculated for four directions: top down, right left, bottom up, and left right. Then, flow extension is checked. If there are at least two flows with extensions greater than a threshold, go for the second step, if not pass to the next pixel. The second step consists in validating the flow according to its length despite variations in the flow extension that may make it a subset of the original flow found at first step. So the integral subset flow is looked for continuation in every direction as respecting the following conditions: (1) the predecessor flow must be subset of the successor flow (2) the integral flow does not turn back towards its starting point (3) the integral flow must remain a straight line. Then, we validate flows longer than a threshold. Each validated flow is approximated by a line segment S characterized by the following 4-vector: starting point $\mathbf{p}$, length l, orientation θ and contrast (c_l, c_r) (average grey level on the left/on the right), $\overrightarrow{S} = [\mathbf{p}\ \theta\ l\ (c_l, c_r)]$. Its *reliability* is defined as the product $\mathcal{E} \times l$ of the extension by the length. The last step consists in validating the junction. A junction combines separate flows at a given point −at least two validated flows (line segments)− . As a *primary variable* P, the junction is characterized by five or seven parameters: $\Delta\theta$ being the angle between any two line segments, $\overrightarrow{P} = \left[\mathbf{p}\ \overrightarrow{S}_1\ \overrightarrow{S}_2\ \overrightarrow{S}_3\ \Delta\theta_1\ \Delta\theta_2\ \Delta\theta_3\right]$.

3 Matching Candidate

When the primitives have been extracted separately from each image, following the method introduced above, candidates have to be prepared for matching.

Each primitive will create a preference list containing its potential mates (the primitives of the other image to be possibly paired). Matching candidates are searched for inside a bounded window the size of which results from a trade off between computation cost and application constraints. The compatibility between primitives is first tested following three constraints on the junction properties : (1) there is at least one common level line between two junctions (2) they must have the same order of area's intensities (3) angles between any two level line flows cannot change by more than 180 degrees. Note that preference lists are thus incomplete (not all primitives in the other image belong to the list) and likely have different number of matching candidates for different primitives. The preferences are set then based on the junction similarity. Two categories are distinguished according to the transformation range in the targetted application.

(1) if the images to match show an important displacement between them, we rather use the junction properties for its similarity $\mathcal{S}$: it is a weighted sum of three separate terms: (1) junction reliability $\mathcal{F}$, (2) region characteristics around the junction $\mathcal{R}$, and (3) geometric invariance $\mathcal{G}$.

$$\begin{aligned} \mathcal{S} &= \alpha\mathcal{F} + \gamma\mathcal{R} + \delta\mathcal{G} \\ \mathcal{F} &= \frac{\sum_i^N \mathcal{F}_i}{N} && \text{with } \mathcal{F}_i \text{ the segment reliability,} \\ & && N \text{ the segment number} \\ \mathcal{R} &= \frac{\sum_{i=0}^N \left\| I(\mathcal{R}_i) - I(\mathcal{R}'_i) \right\|^2}{N} && \text{with } I(\mathcal{R}_i) \text{ the contrast } (c_l, c_r) \\ \mathcal{G} &= \sum_{i=0}^N |\frac{l_i}{l_{i+1}} - \frac{l'_i}{l'_{i+1}}| + \sum_{i=0}^N |\Delta\theta_i - \Delta\theta'_{i+1}| \end{aligned} \tag{1}$$

(2) if the displacement is small (like in limited motion analysis) a classic similarity can be used , as the correlation or the sum of differences.

4 Stables Marriages Matching

After preselecting matching candidates by creating the preference list for each primitive, an algorithm of stables marriages suitably designed for such problems (see algo.1) will be used for actual matching. Let us first explain the stable marriages paradigm ([8], [9]). In this problem, two finite sub-sets M and W of two respective populations, say men and women, have to match. Assume n is the number of elements, $M = \{m_1, m_2, ..., m_n\}$ and $W = \{w_1, w_2, ..., w_n\}$. Each element x creates its preference list $l(x)$ i.e. it sorts all members of the opposite sex from most to less preferred. A matching $\mathcal{M}$ is a one to one correspondence between men and women. If (m, w) is a matched pair in $\mathcal{M}$, we note $\mathcal{M}(m) = w$ and $\mathcal{M}(w) = m$ and ρ_m is the rank of m in the list of w (resp. ρ_w the rank of w in the list of m) . Man m and woman w form a blocking pair if (m, w) is not in $\mathcal{M}$ but m prefers w to $\mathcal{M}(m)$ and w prefers m to $\mathcal{M}(w)$. If there is no blocking pair, then the matching $\mathcal{M}$ is stable.

A reliable algorithm of stable marriages in stereo or motion that is global fitting from local attraction, should fulfill three criteria: stability (i.e. no local

questionning of more global associations), sex equality (i.e. local/global balance of the resemblance to matching) and global satisfaction (i.e. limited amount of local counter run).

Classic stable marriage algorithms (see [8]) of complexity $O(n^2)$ guarantee the stability only. The solution can be such that every primitive has a weak fit. The proposed algorithm *Blocked Zigzag* (*BZ*) (algo.1) meets the three criteria thanks to a novel representation, called *marriage table*, that translates and supplements the preference lists. The *marriage table* is a table with $(n+1)$ lines and $(n+1)$ columns. Lines (resp. columns) frame the preference orders of men, $\{1 \cdots p \cdots N \ \infty\}$ (resp. women, $\{1 \cdots q \cdots N \ \infty\}$). The cell (p, q) contains pairs (m, w) such that w is the p^{th} choice of m, and m is the q^{th} choice of w. Cells can thus contain more than one pair or none. The cell (p, ∞) (resp. (∞, q)) contains the pairs where the woman is the p^{th} choice of the man (resp the q^{th} choice of the woman) but the man does not exist in her preference list (resp. the woman is not in his preference list). A key feature of this table in the "complete list" case is that each line contains all men once and each column contains all women once (see figure 1(a)).

Stable matchings are looked for by scanning this latter array and suitable properties of the solution are associated to the type of scan. Indeed, one advantage of the marriage table is that satisfaction, equality of sex and stability show concurrently in the same representation. A solution with maximum global satisfaction would display matched pairs as close around the origin (table bottom-left) as mutual exclusion allows. More generally the table representation is indicative of a result global satisfaction through the lay out of the selected couples. Intuitively the closer to the diagonal the more balanced treatment. Elements of a pair in a cell close to the diagonal are equally satisfied or unsatisfied, depending on the distance to the origin. Stability gets a graphic translation too in the marriage table.

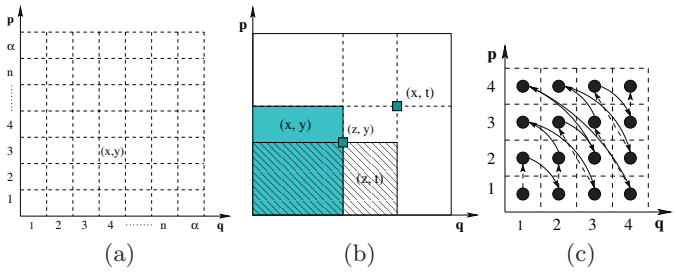


Fig. 1. (a) Marriage table : the pair (x,y), y is the 3^{th} choice of x and x is the 4^{th} choice of y. (b) Blocking situation in marriage table. (c) *BZ* algorithm.

The *BZ* algorithm scans anti-diagonals of the table forward from maximum to minimum global satisfaction but each one is read in swinging from center to sides meaning maximum to minimum sex equality (see figure 1(c)). In each cell, all pairs are accepted for marriage if their components are free. After all

cells have been considered, the table is then revisited up to complete removal of blocking situations as follows: potential blocking pairs are matched upon detection (test according to figure 1(b)) while both blocked couples are broken and complementary elements are freed. To overcome cycles in the assignment the number of rescanning is limited to the population size. Scan directions together with questioning all previous marriages on demand guarantees the better at end. However, *BZ* shows an increase in complexity to $O(n^3)$ due to systematic test added.

Algorithm 1. Blocked zigzag algorithm

```
begin
    while there is a bloking pair and rescan number < population size do
        foreach anti-diagonal, maximum to miminum global satisfaction do
            foreach diagonal, maximum to miminum sex equality back and forth
            do
                foreach pair (m, w) do
                    if m and w are free then
                        Marry m with w
        foreach anti-diagonal, maximum to miminum global satisfaction do
            foreach diagonal, maximum to miminum sex equality back and forth
            do
                foreach pair (m, w) do
                    if (m, w) is blocking pair then
                        Free m and w and their spouse
                        Marry m with w
end
```

5 Eliminating Outliers

BZ organizes the best correspondence possible for each primitive. Still, when corresponding primitives do not exist in either image, the matching couple cannot be else than a mismatch or a missing match, called *outlier*. We draft here the simple method for eliminating outliers. Considering each couple as an "optical flow" and assuming there are only small displacements locally in images, such optical flow gets same length and direction in average. Whence the algo.2:

6 Global Image Transformation

After matching and outliers elimination, couples can be assumed reliable enough to estimating the global transformation between images. We outline here the estimation process in the case of perspective transformation model that requires

Algorithm 2. Outliers elimination algorithm

```
begin
    foreach optical flow in small window do
        foreach x ∈ {angle, length} do
            Find neighbors-optical flows-
            Order neighbors by x
            Compute x between any two close flows
            Order x from min to max
            Find optimal threshold (x histogram)
            Compute x in order (current flow, neighbors)
            if x > optimal threshold then
                Delete the current optical flow
end
```

to estimate 8 parameters. Model is of the form $\mathbf{y} = \mathbf{Xh}$, where h is the parameters column: it follows 2.

$$\begin{bmatrix} x'_0 \\ y'_0 \\ \vdots \\ y'_N \end{bmatrix} = \begin{bmatrix} x_0 & y_0 & 1 & 0 & 0 & 0 & -x'_0x_0 & -x'_0y_0 \\ 0 & 0 & 0 & x_0 & y_0 & 1 & -y'_0x_0 & -y'_0y_0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & x_N & y_N & 1 & -y'_Nx_N & -y'_Ny_N \end{bmatrix} \begin{bmatrix} h_0 \\ h_1 \\ \vdots \\ h_7 \end{bmatrix} \tag{2}$$

And using the least square recursive method in [10], the transformation parameters are given by $\mathbf{h} = (\mathbf{X}^H\mathbf{X})^{-1}\mathbf{X}^H\mathbf{y}$, $\mathbf{h} = \mathbf{X}^{\#}\mathbf{y}$, hence $\mathbf{h}_n = \mathbf{Q}_n\mathbf{X}^H\mathbf{y}_n$, $\mathbf{Q}_n = (\mathbf{X}_n^H\mathbf{X}_n)^{-1}$

7 Experimental Results

In the sequel two series of test results are shown. First one compares the multi pass Hough transform on extracted level-line-segment based primitives, with the stable marriage algorithm run on the same. The difference between original and transformed back images are displayed to that purpose. It is obvious to the naked eye that results are quite comparable, although the BZ Marriages seem to spread errors more all over the picture an in a lesser amount . Each method gets areas where it performs comparatively better − strong primitives orthogonal to the average displacement for Hough and aligned with it for Marriages− . The same phenomenon occurs on all similar images of the type that algorihms were tried on, as long as the transformation to be exhibited is not too complex (limited number of parameters). The present example is extracted from the vision of a car getting out of a parking and involves translations and planar rotations merely. Therefore in a second series of experiments, results of primitive marriages are shown on stereo images extracted from the data base "http://www.gravitram.com/stereoscopic_photography.htm" of monuments. The algorithm again performs qualitatively well despite projections involved. Good news is that large structures are well distinguished, relatively

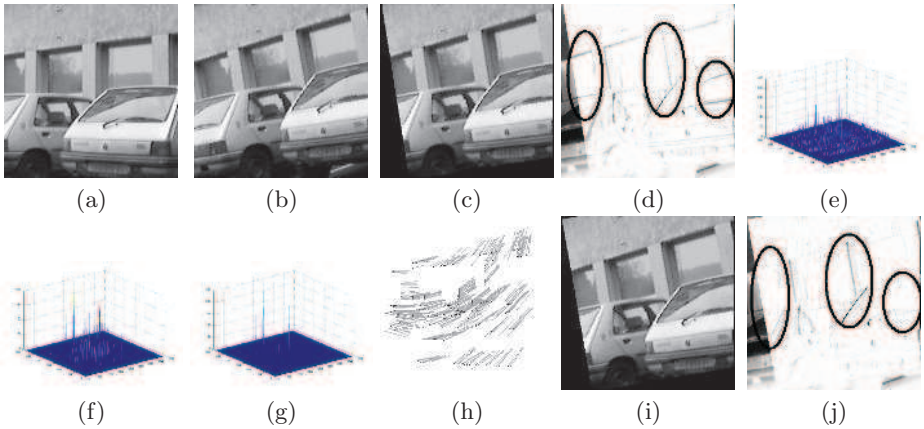


Fig. 2. (a)(b) Original images to match, (c) Affine transformation by Hough: $\alpha = -9.0$, $\psi = 1.0096$, $T_x = 5$ and $T_y = 3$, (d) The difference between (b) and (c), (e) The vote space after Hough 1^{st} round, (f) 2^{sd} round, (g) 3^{rd} round, (h) Matching result by BZ, (i) Affine transformation by BZ: $\alpha = -9.95$, $\psi = 0.9977$, $T_x = 4.10$ and $T_y = 6.73$, (j) The difference between (b) and (i)

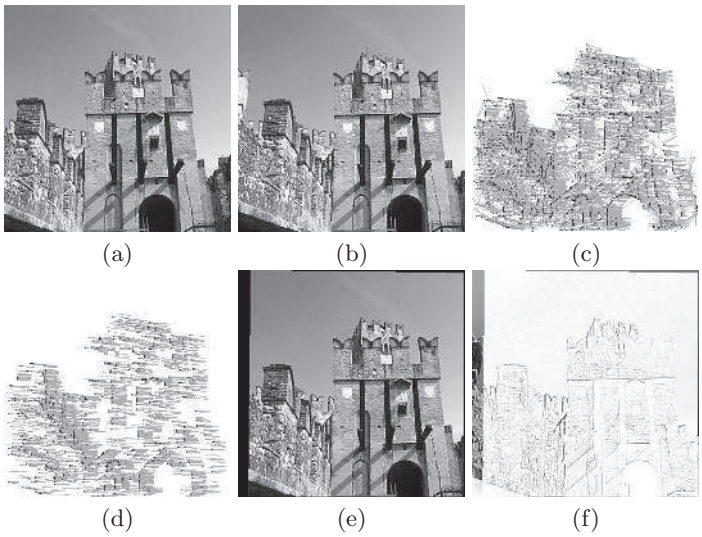


Fig. 3. (a)(b) The stereo images, (c) Matching results by BZ, (d) Result after elimination of outliers, (e) Perspective transformation: $h_0 = 1.018769$, $h_1 = -0.010705$, $h_2 = 20.975390$, $h_3 = 0.013266$, $h_4 = 0.986765$, $h_5 = 2.239246$, $h_6 = 0.000062$, $h_7 = -0.000035$ and $h_8 = 1.000000$, (f) The difference between (b) and (e)

bad news is that it seems to be to the detriment of more tiny details. That would ask further study of the minimization precision vs. the size of considered primitives, and more generally the impact of the features of selected segments (length, contrast, orientation etc.).

8 Conclusion

Stable marriages run comparatively well on image couples or sequences segmented into level line based primitives. Our main applications being car driving and experimental physics (electron microscopy, MRI etc.), they involve images with potential high rate of ambiguities. We thus need to compare results with our Hough technique satisfactorily used so far in a more quantitative way and find explanations why one performs better that the other in which cases. That is our next step in the study, as the whole matching process currently under investigation and drafted in the present paper stands, better than many, the growth of the transform parameter number.

References

1. B. Burg and B. Zavidovique. Pattern recognitions and image compression by means of a time warping algorithm. *ICPR Paris*, Octobre 1986.
2. S. Bouchafa and B. Zavidovique. Stratégie de vote pour la mise en correspondance de lignes de niveaux. *RFIA'04*, January 2004.
3. K. Zemirli, G. Seetharamann, and B. Zavidovique. Stable matching for selective junction points grouping. *IEEE JCIS*, February 2000.
4. V. Caselles, B. Coll, and J. Morel. Topographic maps and local contrast changes in natural images. *International Journal of Computer Vision*, 33(1):5–27, September 1999.
5. P. Monasse and F. Guichard. Fast computation of a contrast-invariant image representation. *IEEE Trans. on Image Proc.*, 9(5):860–872, 1998.
6. N. Suvonvorn, S. Bouchafa, and L. Lacassagne. Fast reliable level-lines segments extraction. *ICTTA*, 2004.
7. S. Bouchafa and B. Zavidovique. Cumulative level-line matching for image registration. *IEEE 12th International Conference on Image Analysis and Processing*, pages 176–180, September 2003.
8. D. Gal and L.S. Shapley. College admissions and the stability of marriage. *American Mathematical Monthly*, 69:9–15, 1962.
9. D. G. McVitie and L. B. Wilson. Three procedures for the stable marriage problem. *Communications of the ACM*, 14,7:491–492, July 1971.
10. Grard Blanchet and Maurice Charbit. *Signaux et images sous Matlab*. HERMES Science Europe Ltd, 2001.

Available online at www.sciencedirect.com
SCIENCE DIRECT®
Real-Time Imaging 10 (2004) 9–22
www.elsevier.com/locate/rti

Time-scale change detection applied to real-time abnormal stationarity monitoring

Didier Aubert[a], Frédéric Guichard[b], Samia Bouchafa[c,*]

[a] INRETS, 2 av. du Gal Malleret-Joinville, 94114 Arcueil Cedex, France
[b] Poseidon Technologies, 3 rue Nationale, 92100 Boulogne, France
[c] Institute d'Elecronique Fondamentale, University Paris Sud, Bt 220, 91405 Orsay Cedex, France

Abstract

This paper presents two robust algorithms with respect to global contrast changes: one detects changes; and the other detects stationary people or objects in image sequences obtained via a fixed camera. The first one is based on a level set representation of images and exploits their suitable properties under image contrast variation. The second makes use of the first, at different time scales, to allow discriminating between the scene background, the moving parts and stationarities. This latter algorithm is justified by and tested in real-life situations; the detection of abnormal stationarities in public transit settings, e.g. subway corridors, will be presented herein with assessments carried out on a large number of real-life situations.

1. Introduction

This paper presents a device developed as part of the European CROwd MAnagement with Telematic Imaging and Communication Assistance (CROMATICA) and Pro-active Integrated systems for Security Management by Technological, Institutional and Communication Assistance (PRISMATICA) projects. Their aim is to assist, through the use of computer imaging tools, public transit operators in their daily tasks, such as incident and anomaly detection. This kind of detection is useful for reacting quickly to events capable of disturbing network management (e.g. entry into forbidden areas, illegal track crossing, counter-flow movements), decreasing user safety (e.g. dizziness, fighting, vandalism, drug deals, unattended objects, overcrowded platforms, person requiring assistance) or decreasing the system's attractiveness (presence of vagrants, beggars, illicit vendors, graffitists).

In this paper, we concentrate on the detection of abnormal stationarities in subway corridors using an image processing device. By abnormal stationarity, we are implying excessive periods of immobility in places such as corridors, where long stays are generally uncommon. Prolonged stationarity may indicate the presence of: a person suffering from illness, a person fallen and unable to rise, a thief waiting for a potential victim, a drug dealer, an illicit vendor, a beggar, a vagrant, an abandoned object or graffiti being painted on a wall.

This paper is organized as follows. We first present our application and its constraints. We emphasize the importance of defining a method that is effective under low contrast and stable under contrast change. In Section 3, we describe a "change-detector" algorithm based on specific primitives: the level lines of the image. As shown in [1–3], this representation is, to a certain extent, contrast-independent, a feature which allows us to design a detection based on image "*geometry*" *rather than* on image "*contrast*". We show that the *intersections or the disappearance* of level lines between images and a specially built "reference" *are indications of changes in the images* (novel/moving objects). We then derive a real-time algorithm from these items. This algorithm does not require any arbitrary threshold especially in the primitive extraction process. The only necessary parameters are: the historical time T, and the occurrence threshold T_o that need to be fixed from the application constraints. In short, these parameters allow us to state: "This event has been present in the images for less than T_o units of time within the last T units of time; a change has therefore transpired."

*Corresponding author.
E-mail addresses: didier.aubert@inrets.fr (D. Aubert), fguichard@poseidon.fr (F. Guichard), samia.bouchafa@ief.u-psud.fr (S. Bouchafa).

1077-2014/$ - see front matter © 2003 Elsevier Ltd. All rights reserved.
doi:10.1016/j.rti.2003.10.001

In Section 4, we demonstrate how to use the same change detector algorithm to detect stationary objects by just adjusting these two parameters. More precisely, we observe that a comparison of the results obtained by applying the change detector algorithm twice with both a long and a short historical time yields a satisfactory instantaneous indication of the stationary objects. A summation of stationarities over time enables us to estimate, at each pixel, a stop duration which is then used to trigger an alarm once a threshold has been reached. In our application, we have considered a 2-min duration as it has been established by monitoring operators. We also point out that the algorithm's structure naturally remedies the possible occlusions of such objects by a moving crowd.

Lastly, we present some experimental work and display the level of performance obtained by applying the system in real-life situations.

2. The aim of this application

In order to detect an incident or abnormality, various means are currently being used by transit network officials. The most common in detecting or at least confirming that an incident has occurred is based on closed-circuit television (CCTV) cameras. However, with this equipment, operators are only able to visualize a few cameras on the site at any given time. Moreover, continuously watching screens is rather ineffectual, fastidious and boring. For all of these reasons, it is useful to build a system able to detect events from all the cameras simultaneously. In this case, only some of the pertinent images are sent on to the operators. As a consequence, operator workload gets reduced and the number of detected incidents rises. By increasing the speed of detection, the actions undertaken are more effective, as is the level of safety.

2.1. Previous vision-related approaches

Various applications require the detection of stationarities (e.g. the detection of people waiting for a green light at intersections, the detection of motionless crowds indicating an overcrowded situation).

The general approach consists of constructing a reference image that represents the normal gray-level (or any other measurement for that matter) state of the background scene. The difference with the current image then shows the non-permanent objects, such as pedestrians (see [4,5]). An analysis of the motion of these objects and even sometimes their shape [5,6] can serve to indicate which ones are stationary. However, motion estimation—as well as shape analysis—is not very straightforward within the context of our application. In [7], the authors propose to discriminate stationarities by considering points that belong neither to the background nor to the estimated instantaneous parts in motion. While not explicitly stated, we can remark from their study the implicit use of two time scales for discriminating changes. Motion detection can be considered as a quick change (small time scale) and stationarities as a change occurring over a given time period (intermediate time scale). Our algorithm will make explicit use of this property, by implementing a time-scale adaptable change detector.

Another original approach [8,9] proceeds in a very different fashion by focusing on vertical head (or body oscillations) during the movement of crowds. The authors propose to capture these oscillations (over time) within the frequency domain. A stationary crowd can thereby be detected by the absence of any such frequency patterns. Unfortunately, this method only detects the global stationarity of the scene and has not been designed to detect isolated stationary objects.

Lastly, these systems exhibit a strict set of hypotheses regarding crowd density, camera location and orientation, in order to avoid occlusions, detect vertical oscillations and, in some cases, make assumptions of a sketchy pedestrian model.

2.2. Problems arising with this type of application

2.2.1. Lighting conditions

The quality of the camera used, the lighting environment, dust and the transmission links all suggest that a system for this type of application must be able to accommodate poor-quality images. Due to a generally low level of light, images tend to display noise and the contrast between a person and the background is often minor. Conversely, some cameras may be positioned close to lights, with the resulting images likely to be partially saturated. Other signal disruptions are generated along the transmission links (e.g. oversized cables, switches, electromagnetic sources). In addition, the system may be subjected to illumination variations (e.g. contrast adjustment of the camera itself, modification to the lighting environment, outdoor scenes).

2.2.2. Frequent occlusions

A corridor ceiling is generally too low to obtain a camera view angle that limits occlusion. Thus, when a stationary person is partially or completely occluded by other people or a crowd, the detection process may encounter difficulty in detecting this person and measuring the corresponding stop duration.

2.2.3. Movements of "stationary" people

Furthermore, a stationary person is rarely completely motionless. In fact, even a person standing still makes motion with his legs, arms, etc., thereby adding complexity to the detection process.

3. Change detector algorithm

This section focuses on the "change detector", which constitutes a key generic piece of the overall algorithm proposed for detecting stationary objects. We assume herein that the video device is observing a fixed scene, with the presence of objects moving/disappearing/appearing. In the following discussion, the term "background" will denote the fixed components of the scene, whereas the other components will be referred to as "novelties". The purpose of a "change detector" is, for a given observation, to decide whether each pixel belongs to the background or to the novelties. The change detector is to be distinguished from the motion detection device in the following manner: the change detector is designed to detect new objects/events present in the scene for less than a fixed period of time, whereas the motion detector will only detect moving objects. While all motion implies change, the converse statement is not always true.

As a consequence, methods based on motion detection (see [10–13]) or on motion estimation [14] cannot be applied directly. The use of such methods to switch from motion detection to change detection would at least entail tracking over time those objects [15] that had experienced motion in the past. This step clearly involves complex operations, in particular in crowded and partially occluded environments.

It should also be pointed out that those methods involving multiple camera set-ups (see e.g. [16]) are, for the time being, not being given consideration by network managers for cost reasons. In the future, however, they could provide an appropriate problem-solving approach.

In response to the change detection problem, the classical method would be to build a reference image representing the background state [7]. When the illumination is constant or known, this reference can be built by averaging (not necessarily in a linear fashion) over time the gray-level values at each pixel [4,9,13,17,18]. The average time interval depends on a duration threshold that distinguishes between new and permanent events. Similarly, depending on the situation, gray level values of the image may be replaced by other measurements, such as spatial gradients, wavelet coefficients, edge maps, etc. The essential point herein is to use a measurement technique that yields constant values over time in the background part for the given images.

Intensity-based measurements are now contrast-dependent, a feature which makes this method sensitive to light changes, especially when measurements are accumulated over time (e.g. several minutes). Suggested approaches (e.g. the use of logarithm differences between two successive images [4], reference images based on gradient amplitude [5]) have been experimented in order to overcome light changes, but so far have met with little success. As pointed out in [7], the natural change in gray-level distributions (or in any measurements based on these distributions) does not enable selecting any universal thresholds for discriminating between background and novelties.

Moreover, constructing the permanent ("reference") situation from quantitative measurements is not an easy task: a simple averaging over time may yield values that are affected by both the moving objects (especially in a crowded scene) and contrast changes. Instead, it is preferable to use the qualitative information at each pixel. For this purpose, we will establish a limited list of characteristics at each pixel and measure their occurrence over time, as an alternative to averaging their values.

The use of edge maps, such as the zero-crossings of the Laplacian or of a Canny-Deriche edge detector, constitutes a major step towards withstanding contrast changes [11,17]. Since "edges" (roughly) correspond to large intensity variations, the presence of an edge at a pixel is more stable than the gray-level value itself. Unfortunately, as usually computed, the selection of such edges is in fact made on the basis of intensity (and derivative) criteria. *This selection is contrast-dependent*; therefore, either it is sensitive to contrast change or it discards the low contrast zone (or both). This is not an intrinsic drawback of Gradient or Laplacian but a problem due to the way they are calculated in the computer vision community.

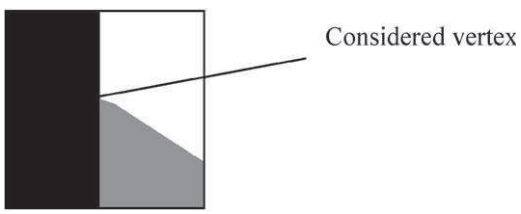
Indeed, two masks M_x and M_y are generally used to estimate the horizontal and vertical gradient components. Quite often, they integrate a smoothing term to filter noise (Prewitt, Sobel, Kirsch masks, for instance). It is the first drawback of the method since this filtering depends strongly on the contrast and alters edges location. Gradient G is, thus, computed by convolving the image with each of the masks:

$$G(\mathbf{p}) = \begin{pmatrix} G_x(\mathbf{p}) \\ G_y(\mathbf{p}) \end{pmatrix} = \begin{pmatrix} I(\mathbf{p}) \otimes M_x \\ I(\mathbf{p}) \otimes M_y \end{pmatrix}.$$

The orientation is then:

$$\theta_{G(p)} = \mathrm{Arctg}\left(\frac{G_y(\mathbf{p})}{G_x(\mathbf{p})}\right).$$

Let us consider, for instance, the following image,



and the following Prewitt masks:

$$M_x = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \quad \text{and} \quad M_y = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}.$$

Around the considered vertex, pixel values are

$$\begin{bmatrix} a & b & b \\ a & c & c \\ a & c & c \end{bmatrix}.$$

Thus, the gradient is

$$G(\mathbf{p}) = \begin{pmatrix} -2b + 2c \\ -3a + b + 2c \end{pmatrix} = 2(c-b) \times \begin{pmatrix} 1 \\ 0 \end{pmatrix} + (b-a) \times \begin{pmatrix} 0 \\ 1 \end{pmatrix} + 2(c-a) \times \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

The terms $(c-b)$, $(b-a)$ and $(c-a)$ correspond to the number of level lines between, respectively, c and b, b and a, c and a. Thus, the computed gradient orientation encompasses geometric information (the vectors) and contrast (the number of level lines). It is the second drawback of the classical method. We can illustrate this result by changing the contrast (variation in a, b, c values) in the example above.

$$\begin{bmatrix} a & b & b \\ a & c & c \\ a & c & c \end{bmatrix} = \begin{cases} \begin{bmatrix} 10 & 15 & 15 \\ 10 & 70 & 70 \\ 10 & 70 & 70 \end{bmatrix} \Rightarrow G = \begin{pmatrix} 110 \\ 125 \end{pmatrix} \Rightarrow \text{Arctg}(125/110) = 0.85 \\ \begin{bmatrix} 5 & 50 & 50 \\ 5 & 90 & 90 \\ 5 & 90 & 90 \end{bmatrix} \Rightarrow G = \begin{pmatrix} 80 \\ 215 \end{pmatrix} \Rightarrow \text{Arctg}(215/80) = 1.22. \end{cases}$$

Even if the geometry is the same, gradient orientation estimated in a traditional way leads to two different results (a difference of 0.37 which roughly corresponds to $\pi/8$) depending on the contrast.

To conclude, we believe that it is first necessary to separate contrast information from geometrical information. Due to the variation over time of the former, the uncertainty is raised as to whether the information available on the latter allows deriving a reliable change detector. This issue is addressed below in a section split into two parts: we first describe the level lines representation of the images that enables separating between contrast (the levels) and geometry (the lines). From this representation, we then derive a simple measurement based on geometry. Secondly, we describe how to combine these measurements in constructing a "change-detector".

3.1. Local and not contrast-based image measurements

In this subsection, we set out to define a simple image measurement that remains stable under contrast change. Such a measurement allows us to discriminate between changes due to object motion and those due to illumination effects.

3.1.1. Observations

Let us begin by assuming a total of N observations I_i of a background-only scene S_c (i.e. the scene without any novelty). These images are obtained by a succession of operations that transforms S_c into a discrete and sampled set of data I_i. Among these operations: illumination conditions, lens smoothing, sampling, contrast adjustment, quantization. Due to this series of operations, the images I_i differ from one another.

As seen in [19–21], we can approximate all these operations by means of sampling followed by a global contrast change. This approximation procedure is of course very rough and is only acceptable for video devices in which the smoothing due to the lens requires a small spatial support. Nowadays, as experimentally proven, this procedure may be considered adequate for certain special applications.

Let us denote the sampled version of S_c, from the same sampling as the image, by S_d; the previous approximation then reveals that all observations I_i can be deduced from S_d through a global contrast change (except for new shadows and spotlighting). The term "global" connotes an intensity value change within the entire image that preserves the relative levels of illumination. In other words, for each i, a non-decreasing function g_i (which models the contrast change) exists such that

$$I_i = g_i(S_d). \tag{1}$$

Since all g_i can differ, a gray-level comparison would not produce any tangible result. Let us introduce at this point the level-lines representation of the image.

3.1.2. The level-lines representation [1–3,22]

Let $I(i,j)$ denote the intensity of image I at pixel location (i,j). Ξ_λ is the level set of pixels with intensity

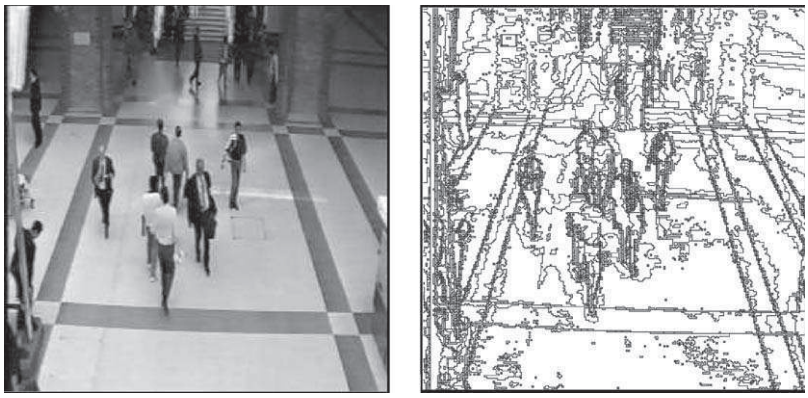


Fig. 1. (left) An image from a London underground sequence and (right) the level lines associated with the image (only 16 level lines displayed).

greater than or equal to λ, i.e.:

$$\Xi_\lambda I = \{\boldsymbol{P} = (i,j), \quad \text{such that } I(\boldsymbol{P}) \geqslant \lambda\}. \tag{2}$$

We shall refer to the boundary of this set as the "λ level line" (see Fig. 1 for an example of level lines). A level line is composed (in the case of discrete images) of a finite number of Jordan curves. As a result of (2), the level sets of an image are included in others, i.e. if $\lambda \geqslant \mu$, then $\Xi_\lambda I \subset \Xi_\mu I$. *Level lines therefore do not intersect*.

3.1.3. Level lines and contrast changes

What is the relationship between the scene S_d and its observation (image I_i)?

If $I = g(S_d)$ with g being a contrast change, then:

$$\{\Xi_\lambda I\}_{\lambda \in \{0,\ldots,255\}} \subset \{\Xi_\lambda S_d\}_{\lambda \in IR}. \tag{3}$$

This would suggest that all level sets of the observed image are indeed level sets of the scene. The operation that switches from the scene to the image can be considered as the simple removal of some level sets and a possible change in their levels. The converse inclusion in relation (3) might not hold true, wherever the contrast change exhibits a flat part, where flat indicates beyond the quantification level. In the worst case, $g(i) = 0 \forall i$, which makes I completely black and therefore only one level set is remaining (that of the entire picture).

What does the intersection of level lines represent?

Relation (3) yields the following:

$$\{\Xi_\lambda I_1\}_{\lambda \in \{0,\ldots,255\}} \subset \{\Xi_\lambda S_d\}_{\lambda \in IR}$$

and

$$\{\Xi_\lambda I_2\}_{\lambda \in \{0,\ldots,255\}} \subset \{\Xi_\lambda S_d\}_{\lambda \in IR}.$$

Since the level sets of S_d are included in others, a contrast change in the scene can create or remove some of the level lines, yet cannot create a piece of level line in one image that intersects a piece of level line in another. Thus, an intersection of level lines is an indicator that something other than a contrast change has occurred between images I_1 and I_2 and, as such, proves to be a useful indicator for a change detection algorithm.

3.1.4. Disturbance effect on level lines

Various types of disturbances may affect the level lines. It is necessary to identify them and to determine the corresponding consequences on level lines in order to adapt our measurement techniques.

The smoothing effect, due to the lens, has already been studied [20]. The impact on the level lines is a displacement of 2 pixels at the maximum with a standard quality lens. Our experiments have shown that such an impact is indeed very low.

Quantification affects the pixel gray level for ± 1 quantization level and thus influences the number of level lines passing through the pixel. This disturbance is mainly visible in homogeneous areas where the level lines fluctuate.

The noise within the image may have two consequences. The holes or peaks in the intensity dimension of the image corresponding to the noise will generate small, isolated level sets. Noise may also locally disturb the direction of the level lines.

Appearing shadows generate additional level lines. While in many video surveillance applications shadows may disturb the detection process, in the present application they can serve to signal the presence of a key object. In some of our experiments for example, the system was able to detect a stationary person behind a pillar due solely to the shadow being cast. When a light bulb burns out, another light suddenly gets turned on or when the illumination changes within the scene due to atmospheric conditions, certain shadows and highlights may appear. In our outdoor applications (i.e. road traffic measurement, Automatic Incident Detection, estimation of vehicle queue length at road junctions), shadows and backlighting are generally discarded due to both their nature [23] and to the fact that such a result appears suddenly at a given location without any previous tracking [24].

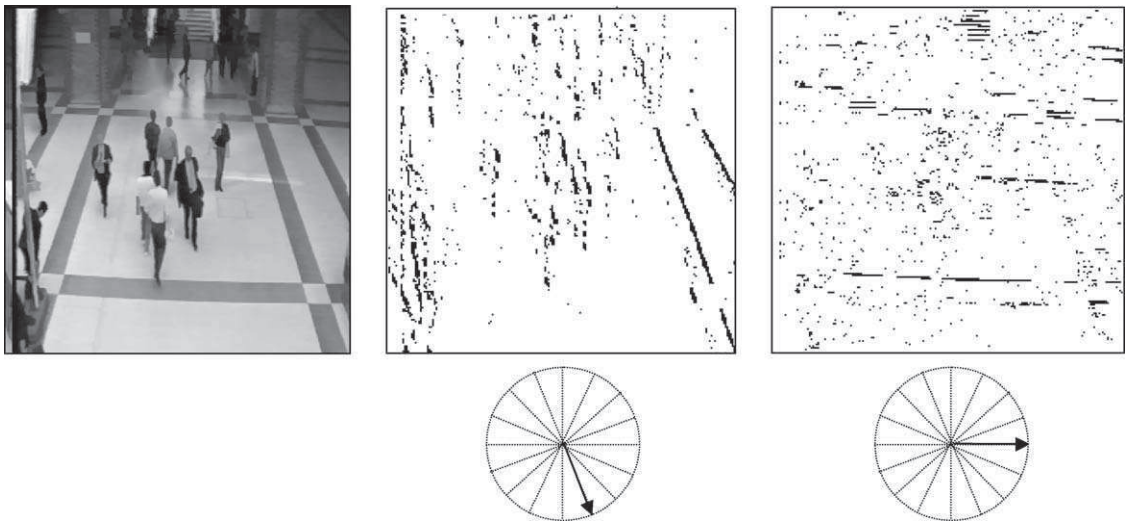


Fig. 2. (left) The original image and (middle and right) display of all pixels on which a level line with the quantized orientation (every $\pi/8$) passes, as indicated by the arrow inside the circle.

3.1.5. Local measurements based on level lines

Working with curves (level lines) is not very convenient in practice, especially for real-time applications. We propose herein to construct at each pixel an "abstract" of the geometry of neighboring level lines. Such a measurement must be local in order to avoid mixing up the background part and the changing part. Since any local measurement based on the level-lines geometry is admissible, we choose to compute the direction of the half-tangent of each level-line passing through each pixel. Consequently, several orientations may be presented for any given pixel; these orientations are then quantized for easier storage and use (see Fig. 2 for an example of such a measurement) and for added resistance to direction change due to noise.

Naturally, we can consider the level-line tangent computation as a different way to estimate gradient orientation. However, it is done in a more suitable way since the geometry of the level line is not affected by global contrast change.

3.2. An algorithm based on reference data

Comparing two observations is more restrictive than comparing an observation directly with the scene S_d. If one observation (piece of level line) is present in one image and not in the other, this may be due to a contrast change. In contrast, if we compare an observation directly with the background of the scene S_d (which contains all the level sets), a new piece of level line would thus be created by means of motion or noise. To enhance change detection, we have decided to build a reference data set that approximates the background of the scene, modulo the contrast. We would like to consider this reference therefore as a kind of union of all the level lines of the observed images. In the absence of any motion and noise, this union will become an approximation of all the level lines of the background part of the scene S_d.

3.2.1. Building and updating the reference data

The reference is to reflect the states of the measurements corresponding to the background. Let us first introduce some of the notation:

- $\theta_0, ..., \theta_{\eta-1}$: η possible quantized directions (e.g. every $\pi/8$).
- $f_t(P, \theta_k)$: a value in $\{0,1\}$ indicating whenever the direction θ_k exists at the pixel P at time t.
- $F_{t \leqslant T}(P, \theta_k) = \sum_{t=1}^{T} f_t(P, \theta_k)$: the number of times the direction θ_k exists for P during period T (time "history").

One very simple way to determine if a local orientation of the level lines passing by each pixel is permanent enough to be introduced into the reference involves calculating its occurrence on a given temporal-sliding window. The update consists of re-computing, at every moment t, the number of occurrences of each direction associated with point P:

$$F_{i \leqslant t}(P, \theta_k) = \sum_{i=1}^{t} f_i(P, \theta_k) = F_{i \leqslant t-1}(P, \theta_k) + f_t(P, \theta_k) \text{ or } m \times F_{i \leqslant t-1}(P, \theta_k) + (1-m) \times f_t(P, \theta_k) \text{ with } m = \frac{t}{t+1}.$$

A direction with a large number (threshold T_0) of occurrences will thus be considered as belonging to the

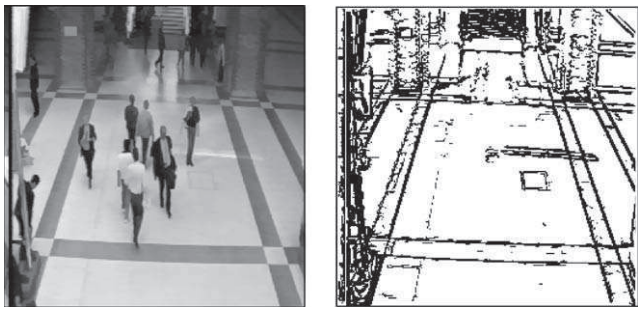


Fig. 3. Example of a reference image: (left) an image extracted from a 200-image sequence and (right) the reference image deduced from the reference data, by means of displaying in black those pixels with at least one orientation that exceeds the occurrence threshold ($T_0 =$ 50 images).

background R. (See Fig. 3 for an example of the reference data obtained.)

$$\left|\begin{array}{ll} \text{If } F_{t\leqslant T}(P,\theta_k) > T_0 & \text{then } R(P,\theta_k) = 1, \\ \text{otherwise} & R(P,\theta_k) = 0. \end{array}\right.$$

3.2.2. Minimal relevance occurrence threshold

Threshold T_o has to be carefully set in order to avoid considering certain moving objects as part of the background. It is generally set on the basis of experience, depending on the behavior of the moving objects (average speed, image rate, etc.). Nonetheless, a natural minimal bound of T_o is provided as a result of the following consideration:

Let us suppose we are looking at a fixed background scene (i.e. no moving objects) plus some random phenomena (such as noise), what then is the minimal threshold T_o such that no randomly created direction will get considered as background? Let us call Pr the probability that a random phenomenon generates a given direction at a given point. The probability that a random phenomenon creates this direction at least s times over T images is then given by the following expression:

$$p(s,T) = \sum_{k=s}^{T} C_T^k Pr^k (1-Pr)^{T-k}.$$

Unfortunately, Pr is not directly computable since we have no a priori knowledge of the cause of the directions observed (noise, moving object, background, etc.). However, what can be directly measured is the probability ($P_{observed}$) that a direction appears on a pixel through a sequence of n images from the visualized scene:

$$P_{observed}^{i,k} = \frac{\text{occurrence of } \theta_k \text{ within the image } i}{\text{number of pixels in } i}.$$

Since $P_{observed}$ overestimates Pr ($P_{observed}$ is, by construction, greater than Pr), we can consider that

$$P_{observed} = \min(\min = P_{observed}^{i,k})_{i=\{0\ldots n-1\}})_{k=\{0\ldots \eta-1\}}.$$

Considering that we have η possible directions per pixel and N_p pixels per image, the expectation of such an event occurring becomes:

$$E = \eta N_p p_{observed}(s,T).$$

Such an expectation must be less than 1; thus, we choose T_o larger than T_{omin}, where T_{omin} satisfies the following:

$$\eta \ N_p \ p_{observed}(T_{omin},T) < 1.$$

Similarly, we cannot choose T_o to be too large, because background elements disturbed by noise would not get incorporated into the reference. This yields a similar upper bound for the occurrence threshold. T_o is to be chosen smaller than $T_{omax} = T - T_{omin}$. We conclude that T_o has to be chosen within the range $[T_{omin,}\ T-T_{omin}]$. In order to be effective, this range must be non-empty, a condition which is always satisfied with a minimal number of images. A high value of T_o produces a reduction in the events (presence of an orientation) stored in the reference. As a consequence, the comparison with a new image will increase the detection rate, while also increasing the number of false alarms. Conversely, a low value will allow for orientations with small occurrences to entering into the reference, causing therefore the detection rates to decrease as well as the number of false alarms. For applications, it is necessary to choose an empirical tradeoff between detection rate and false alarm rate.

In quantitative terms, for an amount of $T = 200(256 \times 256)$ typical images, and with $\eta = 32$, the minimal number of occurrences over time to be considered as background is: $T_{omin} = 20$, which is 10% of the total number of images. With the same number of pixels and directions, and in setting $T_{omin}/T = 3\%$, we would then require a minimal number of approximately 1000 images. This parameter T is often set by the number of the available images provided from the last few minutes, last few hours, last few days, etc. The best initial approach would be to choose this time frame as wide as possible. In any case, it must be wide enough to ensure that the occurrence threshold range is not empty.

3.2.3. Detection

While for building and updating the reference data, we consider all level lines to ensure obtaining as complete a reference as possible, to avoid the "quantification" disturbance during the detection process, only

Fig. 4. (left) An image; (middle) the corresponding detection, without filtering, obtained with reference data built from 200 images and $T_0 = 100$ and (right) the same detection for (white and black) areas larger than 10 pixels.

level lines with at least 2 levels (quantization error + 1) are taken into account. For an observed piece of such level line, two possibilities can arise:

1. The level line exists in the reference, meaning it is not a detection. In fact, it is necessary to take into account the immediate vicinity of the level lines when comparing them to the reference to tolerate a small displacement on the level lines location due to the "smoothing effect".
2. If it does not exist, then a detection is possible. If the piece intersects a level line stored in the reference, then detection is certain. Unfortunately, a detection based solely on the location of a level-line intersection is often, in practice, ineffective. Whenever the background is uniform in intensity up to a certain noise level, such detection is not possible: this is the case, for instance, with roads or corridors. For the other observations, there are two possible causes:
3. either the reference is sufficiently complete, meaning therefore it is a detection, or
4. the reference is incomplete and uncertainty remains as to whether it is a detection or an effect of contrast change.

Given a direction θ at pixel P, we check if this direction occurs in the reference up to both its level of precision and the quantization of the directions stored in the reference (i.e. if $R(P, \theta) = 1$ or not). If such is not the case, then the point is said to be detected (see Fig. 4), which means that either the point is not part of the background or the point is part of the background but the reference is incomplete. Let us also note that a point may exhibit simultaneously several detections, corresponding to several directions that are not in the reference.

3.2.4. Optional filtering process

If necessary, noise can be removed from the detection (see Fig. 4). Morphological filters or median filters can be used for this purpose. If the localization of the detection is to be preserved, we prefer discarding the small isolated detection areas [22] that can be attributed with high probability to noise. We used an empirical setting of 10 pixels, for 256×256 images.

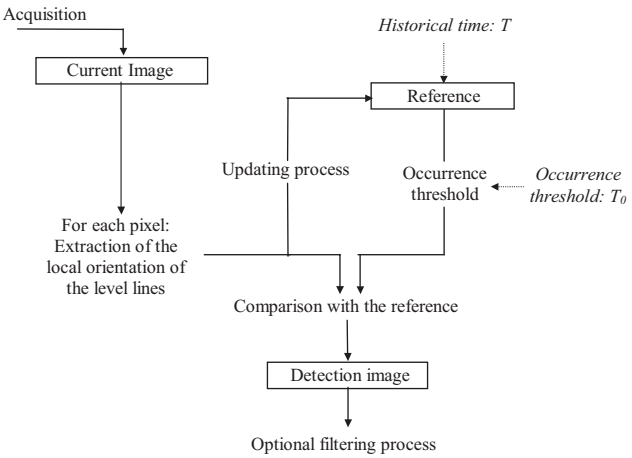


Fig. 5. Diagram of the change detector.

3.2.5. Diagram of the algorithm

See Fig. 5.

3.2.6. Some good properties

In order to demonstrate the robustness of our detection on outdoor scenes and to highlight the benefit of using level-lines measurements, we have conducted the following set of experiments.

The aim of the first experiment is to test the change detection capability of our system on outdoor scene. To ensure that the system is tested under different illumination conditions, we used two image sequences recorded 6 months apart. As shown in Fig. 6, our method yields good results. Of course, in addition to the vehicles, we got shadows and changes in vegetation.

The aim of the second experiment is to demonstrate the capability of encompassing various states of the scene all within the same reference. We built a reference data set from three different image sequences of the

Fig. 6. (top left) Sequence of images used to build the references; (top middle) sequence of images used to test the detection (shot 6 months before the top left sequence) and (top right) detection obtained using our method.

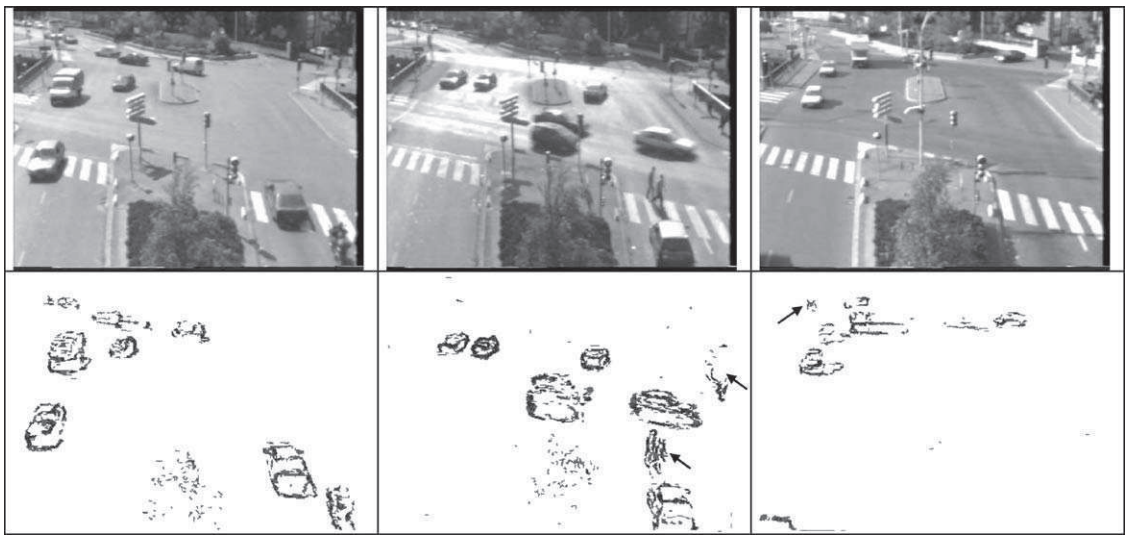


Fig. 7. A single reference data set has been constructed by combining the three sequences from the same intersection. The left and middle sequences were recorded on the same windy day but at different times. The right sequence was taken about 6 months later on a sunny day. In the bottom row, we display (in black) those pixels with level-line directions not contained in the reference data.

same intersection. Two sequences were recorded during a windy day at two different times, while the third one was recorded about 6 months later on a sunny day. As seen in Fig. 7, all of the vehicles and pedestrians indicated by arrows have been well detected. The other detections are due to leaves moving on trees. Thus, the reference data automatically takes into account the various lighting configurations, which confers great stability on the detection with respect to illumination changes. Our algorithm is used herein to control the traffic light at this intersection.

3.2.7. Conclusion

Let us conclude this section by summarizing the main features of the change detector algorithm.

The level-lines representation can be compared with an "edge map" representation, yet with the following distinctions:

- The level-lines representation is a *complete representation* (in that it serves to reconstruct the image, up to a given contrast change). This particular feature allows the algorithm to work even under low contrast once the moving object can be distinguished from the background by a single gray-quantification level.
- Its extraction involves *no threshold or parameters*.
- Its response to a global change of contrast can be described, whereas such tends not to be possible with other "edge map" representations (e.g. zero-crossings of the Laplacian).

The local configuration of the level lines around each point provides a *qualitative* description (which again is obtained without introducing any parameters). This feature enables "counting" the occurrences of each configuration over time in order to create an occurrence-based reference as opposed to an average value-based reference. The pertinence of a configuration is measured herein by its occurrence in time rather than by its ability to surpass a spatial or energy amplitude scale threshold (as is the case with a classical edge map extraction).

In sum, the entire algorithm displays in all just two parameters that we believe are naturally "built in" into

a change detector: the *historical time* T and an *occurrence threshold* T_0. At the end of the process, it is possible to eliminate small areas detected so as to remove noise; this step then adds another optional parameter: the *area threshold*.

4. An abnormal stationary detection system

The extracted level lines could be classified onto one of the following categories: those that belong to the scene background, those that corresponds to moving objects, or to stationary objects.

We propose using the "presence duration" as a mean for discriminating between these three categories. The background is naturally assumed to remain unchanged for a very long period of time. Conversely, the moving objects (a moving crowd) will not yield stable configurations even over short periods of time. In between these two extremes, stationarity is characterized by objects that remain at approximately the same place over an intermediate period of time. This set-up then involves the use of a change algorithm with two time periods: a short one to detect moving objects, and an intermediate one to detect moving objects and stationarities. From this information, we can construct an image containing "stationarity areas". Summing the occurrences of the stationary state at each pixel over time enables estimating the "stop duration". It should be obvious that a stop measurement is possible regardless of the changes in the person/object position inside this entire area or at least part of it. Above a given threshold over this duration, an alarm signal is sent to an operator.

Our system may be summarized by the following diagram (Fig. 8).

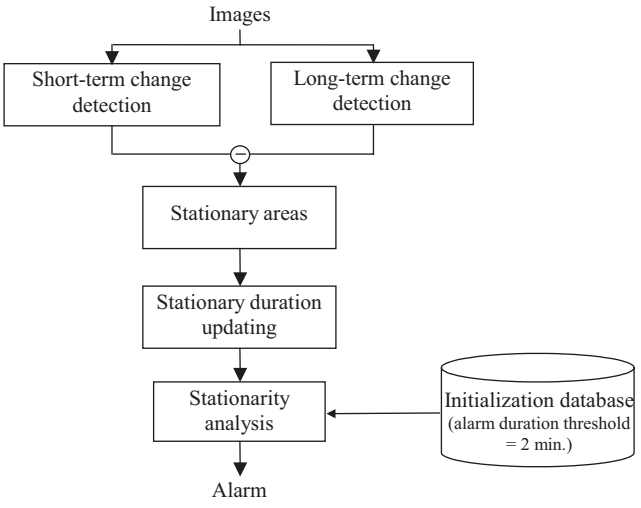


Fig. 8. Diagram of the "abnormal stationarity" detection system devised herein.

Let us now present the algorithm in greater detail:

4.1. Detection of stationary areas

4.1.1. Short-term change detection ("moving parts")

By applying the change detector with a short-term reference R_{short} (computed by taking a sliding temporal window of $T = 0.5$ s and $T_0 = 0.1$ s), areas in the image containing moving objects (representing either a person or an object present at the same place in the scene for less than 0.5 s) are detected.

The result is a binary image representing the short-term changes (M_t):

$$\text{If } \exists\ \theta_k / f_t(P, \theta_k) = 1 \text{ and } R_{short}(P, \theta_k) = 0 \text{ then } M_t(P) = 1, \text{ otherwise } M_t(P) = 0.$$

According to the previous discussion about the occurrence threshold, the number of images considered to create the reference is not high enough to ensure that noise will not disturb the observation.

4.1.2. Long-term change detection ("novelties")

Once again, we could have used the change detector as is, with a higher T and T_0 in order to detect the novelties corresponding to a person or an object present at the same place in the scene for less than 5 min. But instead, we elected to use the previously computed short-term detection in the following way: the long-term reference is only updated at pixels where no motion occurs. At peak hours, the scene is admittedly very crowded (over 90% of the time). By updating references at only the non-moving pixels, we strengthen the quality of the reference data.

Under slight changes, we apply the change detector with a long-term reference R_{long} (computed by taking a sliding temporal window of $T = 10$ min and $T_0 = 5$ min) to detect novelties. The result is the binary long-term change image (N_t):

$$\text{If } \exists\ \theta_k / f_t(P, \theta_k) = 1 \text{ and } R_{long}(P, \theta_k) = 0 \text{ then } N_t(P) = 1, \text{ otherwise } N_t(P) = 0.$$

The lower bound for T and T_0 has been influenced by the objective of sending an alarm within 3 min of detection. A stationary object thus has to be detected (point 1 in the S_t map) for at least 5 min (2 min to detect and 3 min to send the alarm).

4.1.3. Detection of the stationary areas

By removing the motion areas from the novelties, only the people and objects remaining stationary for at least 0.5 s are left. This step is performed by simply subtracting M_t from N_t:

$$S_t(P) = N_t(P) - M_t(P).$$

4.2. Measurement of the stop duration

This functional attribute integrates stationarities over time in order to both compute the stop duration and locate stationary areas.

4.2.1. Estimation of the stationary duration

The estimation of the stationary duration and its update are carried out by an exponential average, in accordance with the following formula:

$$\begin{aligned} &index\_of\_duration(P)_t = (1-\alpha) \times index\_of\_duration(P)_{t-1} + \alpha && \text{if } S_t(P) = 1,\\ &index\_of\_duration(P)_t = (1-\alpha) \times index\_of\_duration(P)_{t-1} && \text{otherwise.} \end{aligned}$$

For a pixel P motionless for the last t images, it can thus be easily verified that

$$index\_of\_duration(P)_t = 1-(1-\alpha)^t$$
$$0 \leqslant index\_of\_duration \leqslant 1,$$

where t is the number of previously observed images and α the value controlling the evolution speed of the "*index_of_duration*" (an index whereby "0" means "nothing present" and "1" means "always present"). α is determined such that "*index_of_duration*" is equal to a threshold T_d when the target maximal stop duration of stationarity (2 min) has been reached. For instance, $\alpha = 1-(1-0.7)^{1/(10\times 2\times 60)} = 0.001003$, when considering a target stop duration of 2 min, a T_d of 70% and a 10 image/s computational rate. T_d is chosen such that any detection is stable (the "*index_of_duration*" may oscillate between T_d and 1.0 without any loss of detection) and such that a detection disappears as quickly as possible when a stationary person/object leaves the scene (the detection ends when the "*index_of_duration*" drops below T_d again).

By computing the stop duration in this manner, we may lose a real detection each time a target area is occluded by a moving crowd. As a consequence, oscillations (detection/no detection) in the detection may be obtained. To overcome this problem, we propose freezing all measurements in the target area as long as the crowd is causing occlusion. In this way, we are able to preserve the estimated duration until the area becomes visible again, at which time the duration increases or decreases depending on the presence or absence of stationarity at the considered location. This procedure adds robustness to the system and stability to the measurements, even when subjected to frequent occlusions.

In the system, the occlusion is characterized by the motion map (M_t). In general, occlusions are caused by a person or a crowd passing in front of the stationary object. This definition includes not only occlusions but also the movements of stationary people, which tend to affect different parts of the body over time:

$$t_{occult}(P,t) = (1-\beta) \times t_{occult}(P,t-1) + (\beta) \times M_t(P).$$

4.2.2. Correction of the duration estimation

By updating the stop duration only when the target is visible, we are able to stabilize the detection and hold the target even when masked by a crowd. In doing so, unfortunately, the estimated stop duration value is lower than the real one in the presence of occlusion. It then becomes necessary to correct the stop duration estimated in order to avoid introducing detection delays. In most instances, this bias can be reduced and even removed: it is simply necessary to increase the evolution speed of the "duration" α according to the number of images for which the computation had not been conducted (t_{occult}). Thus, α is no longer a constant and is to be evaluated for each new image and for each pixel P by means of the following formulation:

$$\alpha(P)_t = 1-(1-T_d)^{1/t'}$$

and

$$t' = t - \min(t_{occult}(P,t),\ 50\%\ t),$$

where $t' \leqslant t$ corresponds to the new number of times an area must be observed motionless before triggering the alarm. However, if we merely note that $t' = t - t_{occult}$ when t_{occult} is close to t, the decision to trigger the alarm will be taken even though the area is seen motionless very infrequently, a situation which may lead to false alarms. To avoid this situation from arising, we have arbitrarily stipulated that a stationary area must be observed for at least 50% t before taking any action. The result is "zero" delay (or actually ± 10 s from our vantage point) when the occlusion rate is less than 50%, but the delay rises for higher occlusion rates.

4.3. Detection of abnormal stationarities

The last functional attribute serves to identify both the potentially abnormal stationarities, based on the stop duration previously measured, and the localization of stop areas. An alarm is then triggered in the event of abnormality.

Each pixel whose stationarity duration exceeds 2 min is labeled as abnormal. Currently, *the detection of just one pixel is sufficient to trigger the alarm* (in practice, the accumulation over time yields very reliable information at each single pixel). The alarm consists of a sound to alert the operator as well as on-screen information to localize the abnormal stationarity on the monitor. This

visual information is conveyed by a red rectangle surrounding the area of stationarity considered to be abnormal.

5. Assessment and results

Our system has been tested on several real-life situations. It is running at more than 10 image/s rate on a Pentium 166 MHz-based PC equipped with a numerization board. During the PRISMATICA project our application (plus three others: intrusion detection, occupancy rate estimation and queue length measurement) was integrated in a small industrial PC (a Celeron 500 MHz) that can simultaneously deal with up to four video sources at a rate of 5 Hz.

Several such devices may be linked, via an Ethernet network, to a supervising system generally located in the control room, which supports the centralized HMI of the system, sends the configuration data and commands.

In order to generate a wide variety of situations (station design, crowd density, location of stationary people/objects, positioning and orientation of the camera), we chose to videotape scenes at the Paris Metro (RATP) station "Havre-Caumartin". In fact, this subway station includes a large number of different corridor configurations (straight and curve, long and short, wide and narrow, light and dark background) and different crowd patterns, due to its proximity to large department stores. In order to increase the variability of the processed scenes even further and demonstrate the transposability of the system to other networks, videotapes supplied by other operators were also used. This approach resulted in more than 224 h of videotape footage and over 400 real-life stationarity situations, some of which been quite difficult to deal with.

In order to assess our system, we began by manually recording (time code of the events, approximate location of the stationarity in the image, and duration of the stationarity) all stationarity events of greater than 2 min. Results obtained by the system were then compared with the recorded data. The differences observed give: the non-detection rate (real-life situations not detected by the system), the number of erroneous detections (detected areas without any real-life justification), and the detection delay. Lastly, the number of stationarities detected versus the total number of real-life stationarities yields the detection rate of the system.

Table 1 summarizes the results obtained from this comparison.

As observed in these results, the system is able to handle the problems affecting this type of application. The results obtained have fulfilled the requirements ($>90\%$ of detection and less than 5 false alarms per day) and demonstrated the ability to cope with some very complex situations (see Fig. 9), which to our knowledge have not been handled by competing systems.

Table 1
Performance of the abnormal stationarity detection system

Number of stationary situations	Number of detections	Non-detections	Erroneous detections	Detection delay (when occlusion $<50\%$)
436	427 (98%)	9 (2%)	0	± 10 s

The non-detections obtained can be explained not only by a low contrast, but also in many instances by a short stationarity duration (around $2\frac{1}{2}$ min) and a very high occlusion rate ($>90\%$). In all cases, except for one, the system detected the stop, but the measured stationarity duration did not reach the threshold above which the alarm is triggered.

The remaining problem then is the delay in detection for occlusion rates above 50%. When this situation arises, a delay in detection can occur and, in the worst case we encountered, may reach approx. 6 min. (This case occurred for a person in the back of the scene completely occluded more than 90% of the time.) In such a situation however, even for an operator to see the person proves to be difficult.

6. Conclusion

The purpose of the automated abnormal stationarity detection system presented herein is to improve the quality of service and crowd management in public transit networks. The levels of performance obtained, even for very complex scenes (very dense crowds, movement of stationary people, low contrast, or subjects positioned in distant locations) surpass the requirements imposed upon operators.

To obtain such performance, we developed the specifications of a new "change" detector based on the representation of images by their level lines. This approach has enabled: producing stable results under illumination changes, detecting objects even under low contrast, and generating a most effective reference data set. This algorithm remains quite generic and has been successfully introduced into some of our other applications, such as traffic monitoring at intersections [24], the detection of counter-flow people motion [25] and queue length measurements. A number of these systems are now available on the market.

Given that operators seem to be showing increasing interest in the system, it is only natural to assume that one day it will be installed in some networks. Such an installation, however, will require a reorganization of the networks' operating mode in order to incorporate

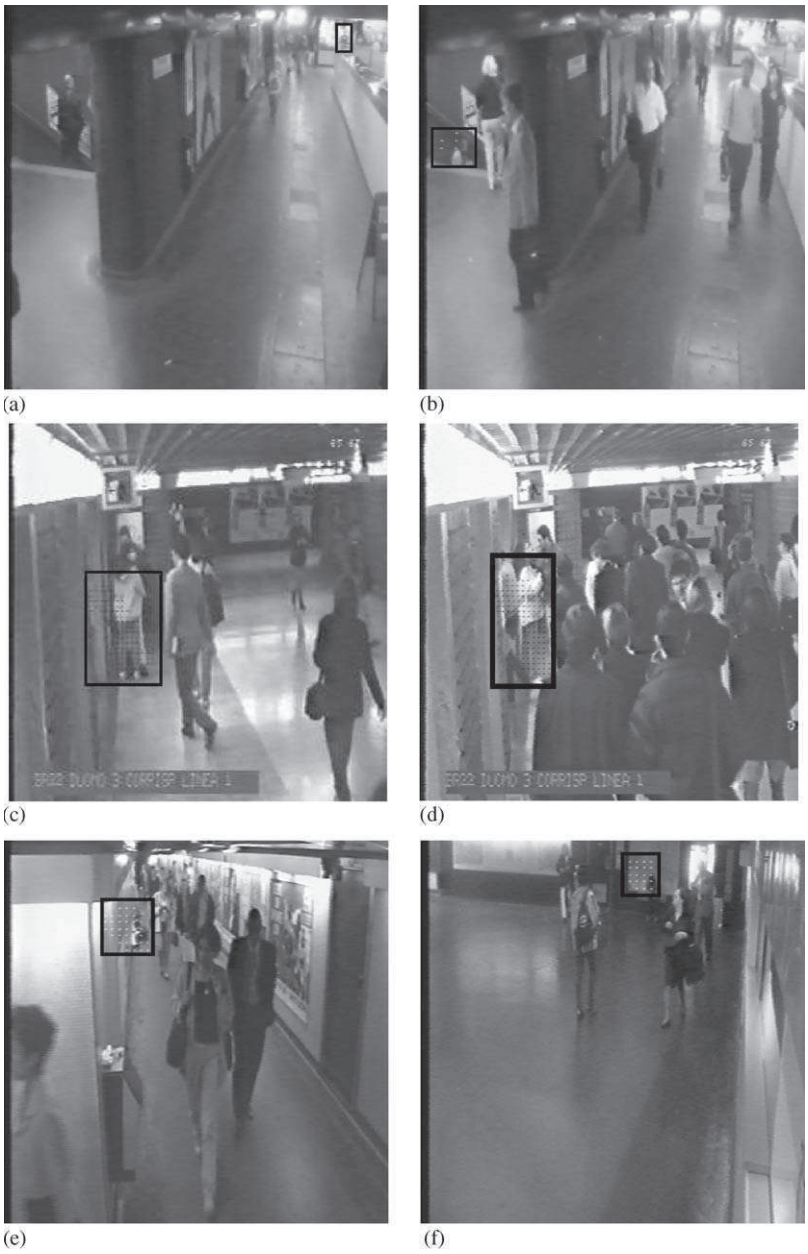


Fig. 9. Examples of abnormal stationary object detection (black squares and dots). (a) A stationary person detected in the back of the scene within a low-contrast and moderately crowded environment; the detection delay is +1 min. (b) The visible part of a seated person detected within a moderately crowded environment; the detection delay is +10 s. (c) Two people detected in a very crowded concourse, hence the occlusion rate is very high for this sequence. (d) The same people after several movements and a displacement without loss of detection at any time; the detection delay is +30 s. (e) An unattended bag detected in a very crowded corridor; the detection delay is +20 s. (f) A seated person detected in the back of the scene in a normally crowded hall; zero detection delay.

the alarm mechanism; among other features, a fast, reliable and effective means of intervention must be implemented.

Acknowledgements

The work presented herein has been undertaken within the scope of the CROMATICA and PRISMATICA projects. It has been supported by an EC grant as part of the 4th and 5th PCRD framework programs. The authors are especially grateful to Ms. B. George and S.-S. Ieng for his valuable assistance during the validation phase of the system.

References

[1] Guichard F, Morel JM. Partial differential equations and image iterative filtering. Tutorial ICIP 95, Washington DC, 1995.

[2] Maragos P. A representation theory form orphological image and signal processing. IEEE PAMI 1989;11(6).
[3] Serra J. Image analysis and mathematical morphology. New York: Academic Press; 1982.
[4] Oscarsson E. TV-camera detecting pedestrians for traffic light control. Proceedings of IMEKO, 1982. p. 275–82.
[5] Reading IAD, Wan CL, Dickinson KW. Developments in pedestrian detection. Traffic Engineering and Control 1995; 538–42.
[6] Reading IAD, Wan CL, Dickinson KW. Detection of pedestrians at puffin crossings using computer vision. Road Traffic Monitoring and Control 1996;23–5.
[7] Gibbins D, Newsam GN, Brooks MJ. Detecting suspicious background changes in video surveillance of busy scenes. IEEE Workshop on Applications of Comp. Vision, 1996.
[8] Davies AC, Yin JH, Velastin SA. Crowd monitoring using image processing. Electronics & Communication Engineering Journal 1995;22–26.
[9] Velastin SA, Yin JH, Davies AC, Vicencio-Silva MA, Allsop RE, Penn A. Automated measurement of crowd density and motion using image processing. Proceedings of the Seventh IEE International Conference on Road Traffic Monitoring and Control, London, UK, 26–28 April 1994. p. 127–32.
[10] Black MJ, Anandan P. A model for the detection of motion over time. ICCV90, 1990. p. 33–7.
[11] Fathy M, Siyal MY. A combined edge detection and back ground differencing image processing approach for real-time traffic analysis. Road and Transport Research 1995;4(3): 1025–39.
[12] Jain R. Dynamic scene analysis. Pattern recognition 2 1985;1: 125–67.
[13] Tekalp AM. Digital video processing. Englewood Cliffs, NJ: Prentice-Hall Inc.; 1995.
[14] Irani M, Anandan P. A unified approach to moving object detection in 2D and 3D scenes. IEEE Transactions on Pattern Analysis and Machine Intelligence 1998;20(6).
[15] Hager GD, Belhumeur PN. Efficient region tracking with parametric models of geometry and illumination. IEEE Transactions on Pattern Analysis and Machine Intelligence 1998;20(10).
[16] Kanade T, Collins RT, Lipton AJ, Burt P, Wixson L. Advances in cooperative multi-sensor video surveillance. DARPA98, 1998. p. 3–24.
[17] Hoose N. Computer image processing in traffic engineering. Taunton: Research Studies Press; 1991.
[18] Siyal MY, Fathy M, Darkin CG. Image processing algorithms for detecting moving objects. ICARCV'94 1994;3:1719–23.
[19] Bouchafa S. Contrast-invariant motion detection: application to abnormal crowd behavior detection in subway corridors. Ph.D. dissertation, Univ. Paris VI-INRETS, 1998.
[20] Caselles V, Coll B, Morel JM. Topographic maps and contrast changes in natural images. IJCV 1999;33(1):5–27.
[21] Guichard F, Bouchafa S, Aubert D, Change detector based on level sets. ISMM2000, Palo Alto, 26–29 June 2000.
[22] Monasse P, Guichard F. Fast computation of a contrast-invariant image representation. IEEE Transactions on Image Processing 2000;9(5):860–72.
[23] Wixson L. Illumination assessment for vision-based traffic monitoring. Proceedings of the International Conference on Pattern Recognition, Track 3: Applications and Robotic Systems, August 1996.
[24] Aubert D, Boillot F. Automatic measurements of traffic flow by image processing: application to the traffic regulation in towns, RTS (Research Transport Safety), No. 62, January–March 1999.
[25] Bouchafa S, Aubert D, Bouzar S. Crowd motion estimation and motionless detection in subway corridors by image processing. IEEE Conference on Intelligent Transport Systems, 1997.