

Analyse d'images en vidéosurveillance embarquée dans les véhicules de transport en commun.

Sebastien Harasse

▶ **To cite this version:**

Sebastien Harasse. Analyse d'images en vidéosurveillance embarquée dans les véhicules de transport en commun.. Traitement du signal et de l'image [eess.SP]. Institut National Polytechnique de Grenoble - INPG, 2006. Français. NNT : . tel-00238440

HAL Id: tel-00238440

https://theses.hal.science/tel-00238440

Submitted on 4 Feb 2008

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE

N° attribué par la bibliothèque

_____

THESE

pour obtenir le grade de

DOCTEUR DE L'INP Grenoble

Spécialité : «Signal, Image, Parole et Télécoms»

préparée au Laboratoire des Images et des Signaux de Grenoble

dans le cadre de **l'École Doctorale «Électronique, Électrotechnique, Automatique, Télécommunications et Signal»**

présentée et soutenue publiquement

par

Sébastien Harasse

le 7 Décembre 2006

Titre :

ANALYSE D'IMAGES EN VIDEOSURVEILLANCE EMBARQUEE DANS LES VEHICULES DE TRANSPORT EN COMMUN

Directeur de thèse : Michel DESVIGNES
Co-directeur : Laurent BONNAUD

JURY

Monsieur	P.Y. Coulon,	Président
Monsieur	L. Chen,	Rapporteur
Monsieur	M. Milgram,	Rapporteur
Monsieur	J.L. Dugelay,	Examinateur
Monsieur	V. Hector,	Examinateur
Monsieur	M. Desvignes,	Directeur de thèse
Monsieur	L. Bonnaud,	Co-directeur de thèse

Table des matières

Introduction **7**
Introduction générale 7
Plan du manuscript 8
Spécificités de la vidéosurveillance embarquée 9
Les variations temporelles de la scène 10
L'adaptabilité du système 12
Limitations dues au caractère embarqué du système 12
Méthode d'analyse vidéo 13

I Analyse de l'état de la caméra **15**

1 État de l'art **17**
1.1 Mesures de netteté 18
1.1.1 Présentation du système optique et du flou 18
1.1.2 État de l'art des mesures de netteté 20
1.1.3 Conclusion sur les mesures de netteté existantes 23
1.2 Estimation de mouvement global apparent 24
1.2.1 Présentation générale du recalage d'images 24
1.2.2 État de l'art du recalage d'images 25
1.2.3 Conclusion sur les méthodes existantes de recalage 29

2 Mesures de qualité du système d'acquisition **31**
2.1 Mesure de la position d'une caméra 33
2.1.1 Points de contours stables 33
2.1.2 Comparaisons entre vues 39
2.2 Mesure de la netteté de l'image 48
2.2.1 Mesures efficaces de la taille des contours 48
2.3 Obstruction du champ de vision par un objet 52
2.3.1 Effets possibles d'une obstruction 54
2.3.2 Visibilité des éléments fixes de la scène 55
2.4 Conclusions sur les mesures de qualité du système d'acquisition 57

3 Applications liées à l'état du système d'acquisition 59
3.1 Autosurveillance des caméras 60
3.1.1 Extraction des états de référence 60
3.1.2 Détection de changement dans les mesures 62
3.2 Aide à l'installation 67
3.2.1 Aide à la mise au point 68
3.2.2 Aide au positionnement 70
3.3 Conclusions sur les applications développées 72

II Analyse du contenu de la scène 73

4 Etat de l'art et architecture proposée pour la détection de personnes 75
4.1 État de l'art de la détection de personnes 76
4.1.1 Systèmes de détection de personnes 76
4.1.2 Modèles d'arrière-plan pour l'extraction des objets d'intérêt 81
4.2 Architecture générale du système proposé 83
4.2.1 Cartes de probabilités 85
4.2.2 Extensions du schéma général 88

5 Extraction d'éléments de la scène 89
5.1 Détection de la teinte chair 91
5.1.1 Modélisation 92
5.1.2 Détection 95
5.1.3 Mesure de la performance de détection 96
5.2 Détection des parties vitrées 96
5.2.1 Quantification de la variation de chaque pixel 99
5.2.2 Application à la localisation de la porte du véhicule 101
5.3 Détection des pixels d'avant-plan par des méthodes classiques 103
5.3.1 Détection du mouvement inter-image 105
5.3.2 Modélisation du fond 107
5.4 Estimation du fond de la scène à un arrêt 111
5.4.1 Caractérisation des séquences d'arrêt du véhicule 111
5.4.2 Présentation générale de la méthode proposée d'estimation du fond à un arrêt 112
5.4.3 Transformation linéaire des pixels d'arrière plan 113
5.4.4 Extraction des pixels d'avant-plan à partir de l'arrière plan estimé . 117
5.4.5 Extension aux vecteurs de caractéristiques 119
5.4.6 Hypothèses multiples sur l'état de l'arrière-plan 122
5.4.7 Évaluation des performances 125
5.4.8 Implémentation efficace du calcul des vecteurs 132
5.4.9 Conclusions sur l'estimation du fond à un arrêt 133
5.5 Conclusions sur l'extraction d'informations bas-niveau 134

6 Détection et suivi de personnes **135**
6.1 Modèle de visage (fusion au niveau pixel) . . . 136
6.1.1 Fusion teinte chair et mouvement . . . 137
6.1.2 Modèle d'ellipse de peau . . . 137
6.1.3 Formalisme statistique . . . 140
6.1.4 Résultats de détection . . . 145
6.2 Modèle de personne (fusion au niveau objet) . . . 147
6.2.1 Description du modèle . . . 147
6.2.2 Cadre Bayesien . . . 149
6.2.3 Estimation des paramètres . . . 150
6.2.4 Résultats de détection . . . 155
6.3 Suivi de personnes . . . 155
6.3.1 État de l'art du suivi de personnes . . . 157
6.3.2 Suivi par prédiction de mouvement . . . 158
6.3.3 Suivi par similarité en apparence . . . 162
6.4 Conclusions sur la détection et le suivi de personnes . . . 164

7 Applications **167**
7.1 Compression sélective . . . 168
7.1.1 Principe . . . 168
7.1.2 Extraction des zones d'intérêt . . . 170
7.1.3 Les différents préfiltrages . . . 171
7.1.4 Conclusion sur la compression sélective . . . 179
7.2 Comptage de personnes . . . 179
7.2.1 Passage d'un segment . . . 180
7.2.2 Performances du comptage . . . 181
7.2.3 Conclusion sur l'application de comptage de personnes . . . 181
7.3 Conclusions sur les applications développées . . . 181

Conclusion et perspectives **185**
Les apports . . . 185
Les perspectives . . . 187

Liste des publications **189**

Bibliographie **190**

Table des figures **201**

Liste des tableaux **205**

Introduction

Introduction générale

Les transports en commun, comme beaucoup d'autres éléments du paysage urbain, sont en constante évolution, apportant toujours plus de confort aux utilisateurs ainsi qu'aux exploitants d'un réseau. Les véhicules d'aujourd'hui sont équipés d'outils technologiques, chargés d'améliorer le service aux voyageurs, qu'il est difficile de percevoir au premier abord. Dans cette univers embarqué, la communication a d'ailleurs un rôle principal. L'information circule à plusieurs niveaux, que ce soit pour permettre au véhicule de se situer géographiquement et d'en informer les voyageurs à chaque arrêt, pour communiquer cette position et d'autres données à un poste central qui gère le réseau de transport, ou pour surveiller l'activité dans le véhicule grâce à un système d'enregistrement vidéo. Il est naturel que le développement des transports se dirige aujourd'hui vers des outils automatisant certaines tâches qui demandaient à l'origine une main d'œuvre conséquente. Du point de vue d'un exploitant de réseau de transport en commun, des besoins nouveaux apparaissent. Tout d'abord en terme d'aide à l'exploitation du réseau, l'exploitant souhaite extraire automatiquement des informations sur l'affluence dans les véhicules en fonction de l'heure et de la date, afin d'optimiser les horaires de passage. D'autre part, le nombre de véhicules du réseau pouvant être important, l'exploitant cherche à faciliter leur installation et leur maintenance.

La vidéosurveillance est aujourd'hui un sujet d'actualité, qui ne cesse pas de faire parler de lui, en bien ou en mal. Toujours est-il que les systèmes de surveillance s'installent de plus en plus dans les milieux urbains, et les transports en commun ne sont pas une exception. En effet, les caméras apparaissent depuis peu à l'intérieur des véhicules. C'est justement ici qu'un lien entre les besoins de l'exploitant du réseau de transport en commun et l'essor de la vidéosurveillance peut être créé, par une analyse automatique des images acquises par ces caméras. En effet, les caméras embarquées dans les véhicules de transport en commun ont pour fonction première de fournir les images de l'activité à l'intérieur du véhicule lorsqu'un incident survient. Elles enregistrent continuellement le véhicule et ses passagers, mais les images acquises ne sont visionnées que lorsque cela paraît nécessaire. Ces données représentent pourtant une source d'information très intéressante pour l'exploitant, à condition de pouvoir les analyser.

C'est dans ce contexte fortement applicatif de vidéosurveillance embarquée que se place cette thèse. Nous allons explorer les diverses possibilités d'exploitation des images de

vidéosurveillance issues des caméras de transport en commun en vue de répondre à certains besoins de l'exploitant d'un réseau. L'analyse des images de vidéosurveillance est séparée en deux catégories, qui forment les deux parties du manuscript :

1. **L'analyse du système d'acquisition** va chercher à caractériser l'état des caméras du véhicule, en terme de netteté de l'image, position de la caméra et visibilité de la scène. Ces informations vont permettre de surveiller automatiquement le bon fonctionnement des caméras, et aussi de faciliter leur installation
2. **L'analyse du contenu de la scène** va extraire des informations sur les éléments de la scène, notamment les passagers du véhicule. La localisation des personnes dans l'image est la première étape vers des applications d'aide à l'exploitation telles que le comptage de personnes ou l'aide à la relecture des bandes vidéo. L'analyse du contenu de la scène nous permet aussi de séparer les pixels de la vidéo en différentes régions afin de compresser la vidéo de manière adaptative, avec une qualité visuelle propre à chaque région.

Les principales contributions de ce travail, qui seront présentées au cours des différents chapitres, sont les suivantes :

- Des mesures caractérisant le système d'acquisition de manière robuste aux changements d'illumination et de contenu de la scène. En particulier, une mesure de netteté basée sur l'estimation de la largeur des contours est obtenue à partir de l'adéquation d'un modèle Gaussien avec bruit sur le profil moyen des contours.
- L'élaboration d'un modèle d'arrière-plan et d'un algorithme de soustraction de fond associé, adapté à des scènes vidéo dynamiques pour lesquelles
  1. l'arrière-plan est fixe mais comporte des petits mouvements locaux,
  2. de forts changement d'illumination peuvent survenir brusquement,
  3. l'arrière-plan peut être caché par des objets d'intérêt sur une grande surface de l'image et pendant une durée importante.
- Un algorithme de détection de plusieurs personnes simultanément robuste aux occlusions partielles. Il est basé sur une combinaison des informations bas niveau d'avant-plan et de teinte chair, dans un modèle définissant les relations géométriques entre ces informations.
- Un algorithme de compression adaptative des séquences vidéo basé sur une segmentation de l'image en régions de teinte chair, intérieur, et extérieur du véhicule. Chaque région est simplifiée par un filtrage plus ou moins fort suivant son importance pour l'exploitation des images en vidéosurveillance.

Plan du manuscript

Après une introduction sur les spécificités de l'analyse d'images en vidéosurveillance embarquée, permettant de mieux cerner les difficultés à surmonter dans ce travail, les différentes études menées seront développées en deux parties.

La partie I est une description des travaux menés sur l'analyse de l'état du système d'acquisition. Des mesures existantes et nouvelles de la netteté des images, de la position du champ de vision et de la bonne visibilité de la scène sont étudiées pour des applications d'aide à l'installation et la maintenance des caméras. Le chapitre 1 présente un état de l'art des méthodes existantes en mesure de netteté et en recalage d'images, deux disciplines qui sont très pertinentes pour la caractérisation de l'état des caméras. Des mesures stables des trois caractéristiques auxquelles nous nous intéressons sont ensuite proposées dans le chapitre 2. Puis les applications développées utilisant ces mesures sont présentées dans le chapitre 3. Notamment, les applications d'aide à la mise au point automatique et d'autosurveillance des caméras seront décrites.

La partie II rassemble les études menées dans l'analyse automatique du contenu de la scène. Il s'agit principalement de la description d'un algorithme complet de détection de personnes, qui a été développé de façon à répondre aux différentes contraintes de notre contexte de vidéosurveillance embarquée. Néanmoins nous nous intéressons aussi à la localisation des parties vitrées du véhicule, qui est intéressante pour certaines applications comme la compression sélective de la vidéo. Dans le chapitre 4, un état de l'art des méthodes existantes pour la détection de personnes est établi. Nous nous intéressons principalement à la détection de visages, ainsi qu'à la détection de piétons, qui sont deux domaines proches de notre contexte, où les méthodes existantes sont pertinentes pour nos images. Une description générale du détecteur de personnes proposé ensuite est aussi présentée dans ce chapitre. Le chapitre 5 traite quant-à-lui de de l'analyse bas-niveau des séquences vidéo. Différents algorithmes vont permettre de caractériser chaque pixel suivant certaines classes considérées. Plus précisément, les algorithmes décrit ici permettent la détection des pixels de teinte chair, d'avant-plan et des vitres du véhicule. La détection des personnes à partir des informations bas-niveau extraites est alors présentée dans le chapitre 6. Deux méthodes sont proposées, l'une basée sur une combinaison des sources d'information de teinte chair et d'avant-plan au niveau des pixels, et l'autre réalisant cette combinaison directement au niveau d'un modèle de personne plus complet. Enfin le chapitre 7 conclut cette seconde partie par une description des applications développées, liées à l'analyse du contenu de la scène. Les applications de compression sélective et de comptage de personnes seront présentées.

Spécificités de la vidéosurveillance embarquée

En premier lieu, nous introduisons ici le lecteur aux spécificités de l'analyse vidéo pour des caméras embarquées dans un véhicule, en proposant une description détaillée des difficultés à surmonter lors de l'analyse de ce type de séquence. Dans un second temps, une première esquisse de la méthodologie à employer sera présentée.

Les séquences vidéos sur lesquelles nous travaillons présentent des particularités importantes par rapport à une application classique de vidéosurveillance. Une étude préliminaire de ces caractéristiques va permettre de cerner les principales difficultés de l'analyse vidéo

dans notre contexte, et de diriger ensuite l'élaboration des outils d'analyse de manière adaptée.

Variations temporelles de la scène

La grande majorité des systèmes de vidéosurveillance utilisent des caméras fixes par rapport à la scène filmée. L'image issue des caméras est alors relativement stable, dans le sens où les changements temporels visibles dans l'image correspondent généralement à des objets d'intérêt. Lorsque la scène n'est pas complètement statique, les variations sont souvent modélisables. Par exemple un changement global de luminosité entre la nuit et le jour est suffisamment lent pour être dissocié facilement des changements intéressants comme le mouvement d'un objet.

Ce n'est pas le cas d'un système embarqué dans un véhicule. La mobilité du véhicule fait perdre à la scène sa stabilité, même lorsque la caméra est fixée. Nous ne pouvons donc pas faire l'hypothèse d'une scène statique et les variations présentes sont très difficiles à modéliser. Néanmoins, le cas étudié est très particulier car une partie de la vue, correspondant à l'intérieur du véhicule, est immobile par rapport à la caméra, ce qui nous permet d'extraire des éléments stables de l'image, sur lesquels peut s'appuyer l'analyse vidéo.

Les variations temporelles se produisant dans une séquence vidéo typique issue d'un véhicule sont de plusieurs sortes. On peut les classer en variations dues au mouvement d'un élément de la scène, et en variations dues à des modifications d'illumination.

Mouvement d'éléments de la scène

Les variations temporelles visibles dans la vidéo sont tout d'abord dues aux mouvements des objets de la scène. Dans notre contexte, il y a deux facteurs principaux qui apportent des variations en mouvement :

- Dans un contexte de vidéosurveillance, les personnes sont souvent présentes devant la caméra. Ici, le mouvement est créé par les passagers qui montent et descendent du véhicule, ou se déplacent à l'intérieur.
- De manière plus spécifique à notre contexte, la mobilité du véhicule induit un mouvement apparent du paysage par rapport à la caméra. Ce mouvement (généralement assimilable à une translation) est visible uniquement à travers les parties vitrées du véhicule. L'autre partie de l'image, correspondant à l'intérieur du véhicule, est fixe par rapport à la caméra.
- Les séquences d'arrêt du véhicule nous intéressent en particulier pour une application de comptage de personnes que nous envisageons. Pendant les arrêts, les régions vitrées du véhicule exhibent là aussi un comportement différent de l'intérieur du véhicule, même si le mouvement y est moins important que lorsque le véhicule roule. A l'arrêt, des mouvements apparents du paysage extérieur par rapport à la caméra sont dus à la montée des passagers exerçant une pression sur les suspensions du véhicule. Le déplacement induit par ce phénomène n'est pas négligeable, de l'ordre de 5 pixels de hauteur pour une image de 560 pixels de hauteur. L'extérieur peut aussi être dyna-

mique, avec des éléments tels que des arbres qui provoquent des petits mouvements locaux.

Modifications d'illumination

Peut être plus difficilement modélisables que le mouvement, les variations en illumination de la scène sont nombreuses dans nos séquences vidéos. Elles sont dues indirectement au caractère dynamique de la scène apporté par les mouvements du véhicule et des passagers, et par le fait que le système doit fonctionner en toutes circonstances.

Effectivement, l'on doit s'attendre à ce que le système de surveillance acquière des images quelque soient les conditions extérieures en terme d'illumination. Premièrement, le système doit fonctionner aussi bien le jour que la nuit, deux périodes pour lesquelles les conditions d'illumination globale de la scène changent de façon drastique. L'éclairage artificiel propre au véhicule apporte lui aussi des variations en lumière, et ne peut pas toujours être considéré comme un changement global sur l'image, ce qui le rend difficilement modélisable.

D'autre part, des changements d'illumination très locaux se produisent sur l'image. Par exemple, un fort éclairement du véhicule par le Soleil provoque des ombres portées dans la scène. Ces ombres portées sont dynamiques car leur position dépend de l'orientation du véhicule par rapport au Soleil, et celle-ci varie régulièrement lorsque le véhicule est mobile. Elles provoquent généralement un changement très fort et local de l'intensité. Il est donc indispensable que les algorithmes développés prennent en compte ces ombres pour que l'analyse de la scène n'en souffre pas.

Dans le même registre, des reflets apparaissent dans certaines zones de la scène, en particulier les vitres du véhicule, ou plus simplement les rétroviseurs.

Enfin, une source de variation d'illumination à ne pas négliger concerne l'ensemble des contrôles automatiques de la caméra. Le système utilise des caméras relativement bon marché, qui possèdent des fonctionnalités classiques qui ne peuvent pas être désactivées. Parmi celles-ci, les contrôles automatiques de gain et de balance des couleurs sont une source importante de variations. La caméra cherche à corriger l'éclairage global de la scène afin que l'image acquise paraisse correctement contrastée. Elle analyse aussi la chrominance de la scène et la modifie de telle façon à ce que la répartition des couleurs dans l'image soit équilibrée. Pour des conditions d'illumination extérieure similaires, les couleurs d'un même objet dans deux images différentes apparaîtront différemment selon le contenu de la scène. En particulier, lorsqu'un passager se trouve face à la caméra, occupant une grande proportion de l'image, il bloque en partie la lumière extérieure arrivant d'habitude sur l'objectif, et ce manque de lumière est compensé par le contrôle automatique de gain. Ces variations d'illumination sont des variations simples et globales de la scène, qu'il est envisageable de modéliser.



(a) (b) (c)

FIG. 1 – Emplacements standard des caméras. (a) : caméra chauffeur, (b) : caméra couloir, (c) : caméra arrière

Adaptabilité du système

Le système de surveillance doit s'adapter à un grand nombre de situations différentes, de façon automatique. L'analyse des images peut apporter une valeur ajoutée au système de surveillance, par des applications d'aide à l'exploitation d'un parc de véhicules, d'aide à la relecture des enregistrements vidéos, ou d'aide à l'installation. Mais son fonctionnement doit être aussi transparent que possible pour l'utilisateur. Cela signifie qu'il doit y avoir un nombre minimum de réglages spécifiques aux outils d'analyse d'images, et que le système doit donc être le plus autonome possible. Les outils d'analyse d'images doivent s'adapter à la scène, pour des positions de caméras variées, et pour les différents contenus possibles de la scène, sans calibration ou réglages délicats préalables.

Selon le véhicule et la position de la caméra, le contenu de la séquence vidéo peut changer fortement. Il y a trois positions standard de caméras pour la surveillance d'un véhicule de transport en commun, qui sont présentées dans la figure 1. Ces positions varient légèrement d'un véhicule à l'autre, et le véhicule lui-même a des caractéristiques qui lui sont propres, telles que la couleur des sièges, la taille des vitres, l'emplacement de la porte, la présence de barres métalliques pour se maintenir, etc. On ne peut donc faire que très peu d'hypothèses sur le contenu des images.

De ce fait, les algorithmes d'analyse d'images qui vont être présentés fonctionnent souvent à partir d'un apprentissage automatique du contenu de la scène.

Limitations dues au caractère embarqué du système

Le caractère embarqué du système de surveillance impose lui aussi son lot de contraintes, qui peuvent influencer le choix des algorithmes d'analyse d'images.

Le système étant placé à l'intérieur d'un véhicule, parmi d'autres équipements, il doit répondre à des contraintes d'encombrement et des contraintes thermiques. Cela influence directement le type de machine sur laquelle sera basé le logiciel. Le choix de la machine s'est tourné vers un PC standard, muni d'une carte mère au format réduit, et de composants dont

la principale qualité est une faible émission thermique. La puissance de calcul s'en trouve limitée, mais l'architecture PC permet de faire évoluer le matériel simplement lorsque de nouveaux composants plus performants, compatibles avec les contraintes techniques et commerciales, seront disponibles. A l'heure actuelle, l'unité centrale dispose d'un processeur à 1Ghz, et de 128Mo de mémoire vive. Vu la quantité des traitements à effectuer et le nombre de caméras fonctionnant simultanément, les capacités réduites du systèmes entrent en jeu comme un facteur majeur dans la conception des algorithmes d'analyse d'images.

L'acquisition des images est réalisée par une carte dédiée. Elle permet l'acquisition à partir de 8 caméras simultanément, avec une cadence totale de 64 images par seconde maximum. En pratique, rarement plus de 4 caméras sont présentes dans le même véhicule. Nous nous attendons donc à une cadence de 16 images par seconde par caméra. Suite à des choix techniques et commerciaux pris lors de la conception de l'enregistreur vidéo, les images délivrées par la carte d'acquisition sont compressées en JPEG. Cette compression est effectuée par la carte elle-même, afin de ne pas surcharger le processeur central. La qualité de la compression est un facteur à ne pas négliger pour l'analyse d'images, car elle dégrade assez fortement les hautes fréquences, ainsi que les couleurs.

Nous ne nous intéressons pas seulement à la partie embarquée du système. L'exploitation des données est aussi réalisable dans un central, commun au parc de véhicules du réseau, dont le rôle est de permettre la visualisation des images en temps-réel à distance, par transmission sans-fil, ainsi que la relecture des enregistrements vidéos lorsque cela est nécessaire. L'analyse d'images peut là aussi apporter une valeur ajoutée, avec des applications aidant à la relecture, ou permettant une compression vidéo adaptée au contenu de la scène. La puissance de calcul et la capacité mémoire disponibles au central sont plus importantes que dans la partie embarquée, ce qui nous permet d'envisager des algorithmes plus lourds en temps de calcul.

Méthode d'analyse vidéo

Les contraintes qui viennent d'être citées, propres à notre contexte de vidéosurveillance embarquée dans un véhicule de transport en commun, nous incitent à développer des outils d'analyse d'images spécifiques. Ces outils doivent permettre l'extraction des informations qui nous intéressent, tout en étant robustes aux différentes perturbations de la scène.

Certaines perturbations peuvent être modélisées. Nous verrons par exemple que les variations d'illumination dues aux contrôles automatiques de gain et de balance des couleurs de la caméra sont modélisables par une transformation linéaire globale des couleurs, dont il est possible d'estimer les paramètres de manière robuste aux autres variations du contenu de la scène.

D'autres perturbations sont trop complexes pour être modélisables, et ne peuvent donc pas être corrigées directement. En particulier, les variations locales d'illumination et le mouvement apparent du paysage extérieur sont des phénomènes qu'il est très difficile d'ex-

traire de la vidéo. La méthode d'analyse vidéo qui sera adoptée pour que les algorithmes soient robustes à ces variations consiste à chercher des éléments de l'image qui sont suffisamment stables vis à vis de ces perturbations, tout en étant suffisamment représentatif de l'image afin que l'information intéressante puisse être extraite sans difficulté. La structure de la scène vide, représentée par les points de contours des éléments fixes du véhicule sera particulièrement utile ici.

Nous verrons aussi que les petits mouvements de la scène lors des arrêts du véhicule, qui doivent être dissociés des mouvements plus importants des objets d'intérêt, sont difficilement modélisables, mais qu'il est possible de représenter l'image dans un autre espace où ces petits mouvements perturbent beaucoup moins les algorithmes.

Au final, les trois types d'approches face aux perturbations de la scène consistent donc en :

1. la modélisation du phénomène et l'estimation de ses paramètres en vue de comprendre sa contribution dans l'image. C'est le cas de la correction de balance des couleurs, utilisée pour l'extraction des pixels d'avant-plan à un arrêt du véhicule.
2. la simplification de l'image, amenant une perte d'information qui annule l'effet de la perturbation, mais laisse présente l'information intéressante que l'on souhaite extraire. Cette méthode est utilisée pour la caractérisation de la position de la caméra.
3. le passage de l'image dans un espace différent, qui est invariant à la perturbation, mais qui n'impose pas une réduction de l'information dans l'image. Ce sera le cas pour l'extraction des pixels d'avant-plan à un arrêt du véhicule.

Première partie

Analyse de l'état de la caméra

Chapitre 1

État de l'art

Sommaire

1.1 Mesures de netteté 18
1.1.1 Présentation du système optique et du flou 18
1.1.2 État de l'art des mesures de netteté 20
1.1.2.1 Modèle de PSF 20
1.1.2.2 Mesures classiques de netteté 21
1.1.3 Conclusion sur les mesures de netteté existantes 23
1.2 Estimation de mouvement global apparent 24
1.2.1 Présentation générale du recalage d'images 24
1.2.2 État de l'art du recalage d'images 25
1.2.2.1 Données considérées pour le recalage 25
1.2.2.2 Mesures de similarité 26
1.2.2.3 Modèles de transformation 28
1.2.2.4 Estimation des paramètres de la transformation 28
1.2.3 Conclusion sur les méthodes existantes de recalage 29

La première partie des études menées dans notre contexte de vidéosurveillance embarquée dans un véhicule de transport en commun concerne la caractérisation du système d'acquisition. Nous souhaitons développer des outils qui permettent au système de surveillance de connaître son état, à partir des images acquises. L'état du système concerne tout ce qui a un rapport avec le réglage des caméras, sans s'intéresser au contenu de la scène. Il s'agit tout d'abord de caractériser la position de la caméra par rapport à son environnement, c'est à dire d'extraire une information à partir des images qui soit relative au champ de vision filmé, et de pouvoir juger cette information comme correcte ou non. Il s'agit ensuite de caractériser la qualité des images acquises, selon deux critères qui sont la netteté et la bonne visibilité de la scène. La netteté des images peut varier selon la qualité de la mise au point effectuée à l'installation, ou suite à un déréglage inattendu. D'autre part, par bonne visibilité, on entend que la scène ne doit pas être occulté par un

objet gênant. Trois critères vont donc nous permettre de caractériser l'état du système d'acquisition :

1. Le champ de vision de la caméra
2. La netteté des images
3. La bonne visibilité de la scène

Pour chacun de ces critères, on souhaite mesurer des caractéristiques pertinentes de la vidéo. Les applications visées sont alors de deux types.

- Tout d'abord, on souhaite réaliser des outils permettant au système de s'autosurveiller, c'est-à-dire de savoir lorsqu'une caméra est déréglée. Pour cela, il faut que les mesures soient suffisamment stables dans le temps en l'absence de déréglage, et suffisamment corrélées à chaque critère pour qu'une détection de changement dans les mesures soit possible.
- Le second type d'application concerne l'aide à l'installation des caméras, et permettra à un opérateur humain d'installer le système de surveillance plus simplement dans le véhicule, grâce à une mesure en temps réel des deux premiers critères.

Dans ce chapitre, nous étudions les méthodes existantes pour deux problèmes d'analyse d'images qui vont nous concerner de près. Le premier problème est celui de la mesure de la netteté (ou du flou) dans une image. Quant-au second problème, il concerne le problème du recalage d'image, qui se rapproche de notre problème de comparaison entre deux vues différentes d'une même scène.

1.1 Mesures de netteté

1.1.1 Présentation du système optique et du flou

La figure 1.1 illustre un système optique de base, composé d'une seule lentille. Un point P de la scène se situant à une distance u de la lentille se projette en un point P' de l'autre côté de la lentille et à une distance v de la lentille. Les distances u et v sont liées par la loi de Lens :

$$\frac{1}{u} + \frac{1}{v} = \frac{1}{f} \tag{1.1}$$

où f est la distance focale. La mise au point d'un système optique correspond au réglage de la distance v entre le capteur et la lentille de telle façon à ce que le point P' appartienne au plan du capteur.

Lorsque la mise au point n'est pas parfaite, le point P se projette sur le capteur en une tache circulaire I, et l'image paraît floue.

En réalité, les points P de la scène sont *a priori* à des distances u variées, suivant la profondeur des objets. En pratique, la mise au point consistera à obtenir une projection correcte sur le capteur pour les points de la scène qui forment le sujet. Les imperfections de l'œil font que le sujet peut paraître net même lorsque ses points ne se projettent pas

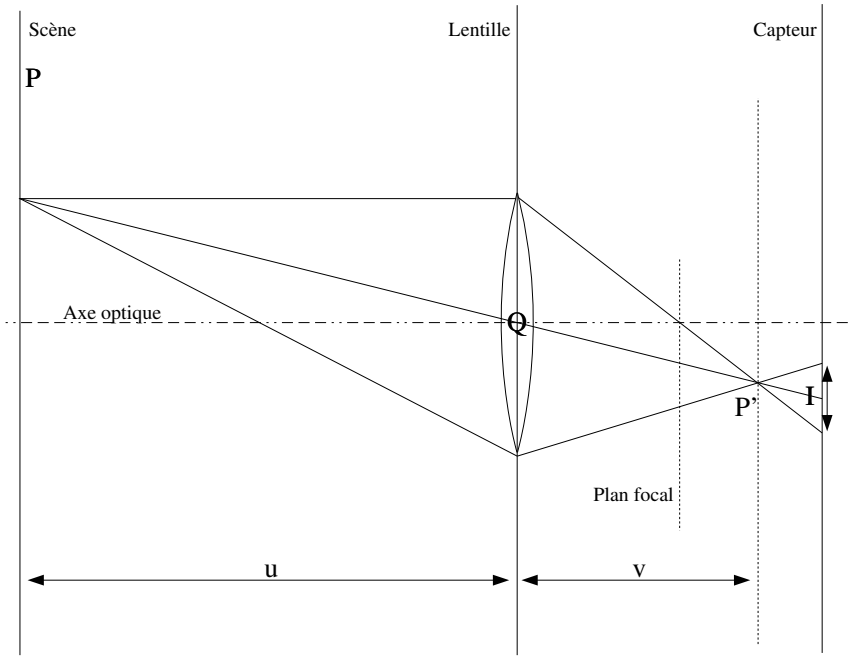


FIG. 1.1 – Schéma d'un système optique

exactement sur le capteur. Pour une mise au point donnée, la gamme des distances u pour lesquelles l'œil voit l'image nette est appelée la profondeur de champ, et dépend principalement de l'ouverture de l'objectif. Pour notre application, on fera l'hypothèse que les objectifs sont tels qu'un changement de mise au point provoque un même flou sur toute l'image. Cela nous permet d'envisager des mesures de netteté globales sur toute l'image.

1.1.2 État de l'art des mesures de netteté

La netteté d'une image est une caractéristique qui a été principalement étudiée dans la littérature pour des applications de mise au point automatique (ou *autofocus*) de caméra. L'objectif d'une caméra forme une image dont les points sont plus ou moins flous en fonction de la distance de chaque objet et des paramètres de mise au point. Le principe de l'autofocus est de faire varier automatiquement la distance entre le centre de l'objectif et le capteur jusqu'à ce que le sujet de la scène soit net. Les systèmes d'autofocus les plus performants de nos jours sont les systèmes à contraste de phase des appareils photo reflex. Ils utilisent deux vues de la scène très légèrement différentes, obtenues grâce à un système de miroir. La différence entre les images des deux vues dépend de la profondeur du sujet, qui peut alors être estimée au travers d'une mesure de contraste. L'appareil règle finalement la mise au point directement en fonction de cette distance.

L'apparition des appareils photo compacts numériques a permi le développement de nouveaux systèmes d'autofocus, basés directement sur une analyse de l'image acquise par les capteurs CCD. En effet, sur ce type d'appareil, l'acquisition de l'image par le capteur est réalisée en permanence, par un obturateur électronique. Cela permet en outre à ces appareils de posséder un mode de visée sur l'écran ainsi qu'un mode d'acquisition vidéo. Il est alors bien plus économique de réaliser l'autofocus directement sur la photo capturée par les CCD, que d'ajouter des composants supplémentaires dédiés à la mise au point. La différence fondamentale par rapport aux systèmes à contraste de phase est que la mise au point doit ici être réalisée à partir d'une image monoscopique, ce qui rend beaucoup plus difficile l'estimation de la profondeur du sujet.

Des méthodes d'analyse d'images permettent d'extraire, directement à partir du contenu de l'image, une mesure dépendante de la netteté. Selon le type d'application, on pourra chercher à obtenir une mesure absolue, indépendante du contenu de l'image, ou non. Pour une application de mise au point automatique, la mesure de netteté s'effectue pendant un temps très court, pour des images dont le contenu est donc très similaire. Le problème de la mise au point automatique revient à un problème d'optimisation des paramètres de la caméra en fonction d'une mesure, et ne nécessite donc pas que la mesure soit indépendante du contenu.

1.1.2.1 Modèle de PSF

Comme illustré sur la figure 1.1, un point de la scène se projette sur le capteur en une tache dont la taille dépend de la qualité de la mise au point. La forme de cette tache est modélisée par une fonction que l'on appelle fonction d'étalement ou PSF (*Point Spread*

Function). L'image I reçue sur le capteur est alors la convolution de l'image nette I_0, qui serait obtenue si la mise au point était parfaite, par la PSF f_{PSF} :

$$I = I_0 * f_{PSF} \tag{1.2}$$

Afin de caractériser la netteté de la caméra, on souhaite estimer les paramètres de PSF à partir de l'image finale I. Une application possible découlant d'une estimation correcte de la PSF est la restauration d'images, qui consiste à retrouver l'image nette I_0 par filtrage inverse. Nous ne nous intéressons pas ici à ce type d'applications de restauration, mais plus simplement à la caractérisation de la netteté de l'image, qui ne nécessite pas de connaître les paramètres exacts de la PSF. Toute mesure sur l'image qui est fortement corrélée aux paramètres de la PSF suffit à caractériser la netteté.

Pour un système d'acquisition disposant d'une seule lentille, de forme circulaire, comme sur la figure 1.1, la PSF est une tache circulaire d'intensité constante. Dans la pratique, les pertes d'énergie autour du centre de la PSF font que l'on assimile généralement la PSF à une fonction 2D gaussienne centrée circulaire :

$$f_{PSF}(\boldsymbol{k}) = \frac{1}{2\pi|\Gamma_{PSF}|^{1/2}} \exp\left(-\frac{1}{2}\boldsymbol{k}^T\Gamma_{PSF}^{-1}\boldsymbol{k}\right) \tag{1.3}$$

avec $\boldsymbol{k} = \begin{bmatrix} x \\ y \end{bmatrix}$ le vecteur de coordonnées dans l'espace image et Γ_{PSF} la matrice de covariance de la PSF, qui ne contient en réalité qu'un seul paramètre scalaire :

$$\Gamma_{PSF} = \begin{bmatrix} \sigma_{PSF}^2 & 0 \\ 0 & \sigma_{PSF}^2 \end{bmatrix} \tag{1.4}$$

1.1.2.2 Mesures classiques de netteté

Les mesures classiques de netteté cherchent à caractériser l'effet de lissage de l'image induit par la convolution de l'image nette I_0 par la PSF. On compte deux grandes familles dans la conception des mesures de netteté. L'*analyse de focalisation* est basée sur une caractérisation globale réalisée sur l'ensemble ou une partie de l'image, tandis que l'*analyse de défocalisation* cherche à mesurer le flou en estimant les paramètres du modèle de PSF.

Analyse de focalisation Parmi les mesures de focalisation, on compte les mesures différentielles, opérant sur le gradient ou le Laplacien de l'image. La PSF distribue l'intensité d'un pixel sur les pixels adjacents, et provoque ainsi un lissage des niveaux d'intensité et par conséquent une baisse des mesures de différence d'intensité entre points voisins. De nombreuses mesures de netteté basées sur cette idée ont été proposées et étudiées dans la littérature. Parmi celles-ci, le critère de Tenengrad [SSV+97] [NPA01] correspond à la recherche de l'image la mieux focalisée par une maximisation de la magnitude du gradient.

Ce critère est calculé comme :

$$F_{tenengrad} = \sum_x \sum_y \|\boldsymbol{g}(x,y)\|^2 \quad \text{pour} \quad \boldsymbol{g}(x,y) > T \tag{1.5}$$

avec $\boldsymbol{g}(x,y)$ le vecteur gradient, obtenu par convolution de l'image avec deux filtres passe-haut (horizontal et vertical) tel que Sobel, et T un seuil permettant de ne prendre en compte que les magnitudes les plus importantes.

Brenner [BDH+76] proposait une mesure similaire, mais basée sur les différences d'intensités à une distance de deux pixels, sur les lignes verticales de l'image :

$$F_{brenner} = \textstyle\sum_x \sum_y |I(x,y+2) - I(x,y)|^2 \quad \text{pour} \quad |I(x,y+1) - I(x,y)| \geq T \tag{1.6}$$

De même, l'énergie du gradient de l'image est calculée comme suit :

$$E = \frac{1}{N} \sum_x \sum_y \|\boldsymbol{g}(x,y)\|^2 \tag{1.7}$$

avec N le nombre de pixels dans l'image. Cette mesure est équivalente au critère de Tenengrad sans seuil, comme proposé par Krotkov [Kro87].

Le contraste dans l'image est aussi une information qui dépend de la quantité de flou. Plusieurs définitions du contraste sont possibles [TLX00] [FGH01]. La forme privilégiée pour la mesure du contraste est la forme énergétique, mesurée globalement sur toute l'image par la variance des intensités :

$$C = \frac{1}{N} \sum_x \sum_y (I(x,y) - \bar{I})^2 \tag{1.8}$$

avec $\bar{I}$ la moyenne des intensités dans l'image.

L'effet de lissage de l'image introduit par une mauvaise mise au point entraîne comme effet une perte d'information, que l'on peut mesurer par l'entropie, comme dans les méthodes proposées par [SSV+97] ou [Yeg99]. Plus la mise au point sera correcte, plus l'entropie sera élevée. On mesure l'entropie de Shannon en fonction de la fréquence d'apparition p_k de chaque niveau de gris k, parmi les K niveaux de gris possibles :

$$H = -\sum_{k=1}^{K} p_k \log_2 p_k \tag{1.9}$$

La fréquence p_k est obtenue en calculant l'histogramme de l'image.

Enfin, l'autocorrélation du signal a aussi été utilisée pour mesurer la netteté d'une image. Effectivement, l'étalement de l'énergie d'un point sur les points voisins provoque un rapprochement des valeurs voisines, que l'on peut mesurer par une mesure de corrélation. Vollath [Vol88] [SSV+97] a été l'instigateur des mesures de netteté basées sur la corrélation. Parmi celles-ci, la mesure du gradient de l'autocorrélation de l'image semble délivrer d'excellentes performances en microscopie génétique :

$$V = \sum_x \sum_y I(x,y) I(x+1,y) - \sum_x \sum_y I(x,y) I(x+2,y) \tag{1.10}$$

L'ensemble de ces méthodes basées sur l'analyse de focalisation ont le défaut d'être très sensibles au contenu de la scène. Elles dépendent effectivement de la qualité de la mise au point de la caméra, mais aussi des objets de la scène et de l'illumination. Une méthode qui est beaucoup plus robuste aux variations de la scène consiste à estimer directement les paramètres de la PSF. Ces paramètres doivent être mesurés sur des points caractéristiques, isolés dans l'image, où la convolution par la PSF a un effet bien reconnaissable. Les points de contours sont justement adaptés à l'estimation de la PSF.

Analyse de défocalisation L'analyse de défocalisation tente d'estimer les paramètres de la PSF à partir des images. Une mauvaise mise au point provoque un étalement de l'énergie en chaque point sur les points voisins. Ce phénomène est particulièrement visible au niveau des contours, qui s'élargissent en fonction de la quantité de flou.

Pentland et Grossman [Pen87], [Gro87] firent les premiers pas dans l'analyse des contours flous en montrant que le laplacien $\triangle I$ de l'image et le paramètre d'étalement σ de la PSF gaussienne sont reliés par la relation :

$$\ln\left(\frac{b}{\sqrt{2\pi}\sigma^3}\right) - \frac{x^2}{2\sigma^2} = \ln\left|\frac{\triangle I(x,y)}{x}\right| \tag{1.11}$$

où b est l'amplitude du contour et (x, y) les coordonnées centrée en un point de contour avec un axe Ox perpendiculaire à celui-ci. Les paramètres b et σ sont obtenus par régression linéaire en x^2.

Les approches basées sur l'analyse de défocalisation peuvent être séparées en deux classes. Les approches de type Fourier travaille dans le domaine fréquentiel, et s'opposent donc aux autres approches travaillant dans le domaine spatial.

Concernant les approches fréquentielles, [Wei94] propose une mesure du paramètre d'étalement de la PSF en vue de l'estimation de la distance d'un objet (problème communément appelé *depth from defocus*). La méthode de [Wei94] nécessite au minimum deux acquisitions d'image avec des paramètres de caméra différents. Les paramètres de la caméra pour les deux acquisitions sont supposés être connus à l'avance, par l'intermédiaire d'une calibration de la caméra. Il ne s'agit donc pas ici d'une mesure qui puisse s'appliquer à la caractérisation de la netteté pour un contexte comme le notre, où une calibration préalable n'est pas permise.

Parmi les approches spatiales, [MDWE02] propose une mesure très rapide de la taille des contours, qui ne considère que les contours verticaux de l'image, et est basée sur la recherche des extrema de l'image lissée à proximité des points de contours. [RRPP02] calcule l'exposant de Lipschitz sur les contours les plus forts, et montre qu'il existe une relation entre cet exposant et le paramètre d'étalement d'un modèle de PSF (variance pour une PSF gaussienne, rayon pour une PSF circulaire, ...).

1.1.3 Conclusion sur les mesures de netteté existantes

Dans les méthodes existantes permettant de mesurer la netteté, celles appartenant à la famille de l'analyse de défocalisation nous concernent beaucoup plus, car notre contexte

impose une très bonne invariance au contenu de l'image. D'autres applications comme l'autofocus d'un appareil photo analysent la scène pendant un temps suffisamment court pour qu'on puisse supposer une variation très faible du contenu. Les applications que nous souhaitons réaliser demandent soit une mesure robuste, soit une mesure rapide. Nous proposerons donc deux mesures, l'une basée sur l'estimation des paramètres de la PSF gaussienne, comme proposé dans l'état de l'art, et l'autre qui est une approximation de la première et peut traiter les images des séquences avec une cadence rapide.

1.2 Estimation de mouvement global apparent

Toujours dans l'optique de mesurer les caractéristiques du système d'acquisition, nous nous intéressons à l'estimation d'un mouvement apparent 2D du champ de vision provoqué par un changement d'orientation d'une même caméra. Nous verrons dans le chapitre suivant que l'on peut effectivement caractériser la position du champ de vision d'une caméra à partir d'informations sur une position de référence et d'une mesure de distance entre cette référence et la position courante. Ce problème de caractérisation de la position d'une caméra est donc fortement lié au problème plus général du recalage global entre deux images, que l'on rencontre dans d'autres contextes tels que l'imagerie médicale (recalage entre acquisitions IRM) ou l'imagerie satellitaire (recalage entre vues locales du terrain pour former une vue d'ensemble). Commençons donc par un aperçu des méthodes existantes en recalage d'images.

1.2.1 Présentation générale du recalage d'images

Le problème du recalage entre deux images I_1 et I_2 consiste à chercher les paramètres optimaux $\boldsymbol{m}$ d'un modèle de mouvement $f_{\boldsymbol{m}}$ qui minimise une erreur globale entre la seconde image et la première image compensée en mouvement :

$$\boldsymbol{m} = \underset{\boldsymbol{m}\in\mathcal{M}}{\arg\min} \sum_{\boldsymbol{k}\in\mathcal{D}} \left(I_2(\boldsymbol{k}) - I_1(f_{\boldsymbol{m}}(\boldsymbol{k}))\right)^2 \tag{1.12}$$

avec $\mathcal{M}$ le domaine des paramètres possibles pour $\boldsymbol{m}$ et $\mathcal{D}$ l'ensemble des coordonnées dans l'image. Le recalage d'images est un outil dont le domaine d'application est plus vaste que celui de l'estimation d'un mouvement de caméra. En imagerie satellitaire, il est souvent nécessaire de recaler plusieurs photographies pour générer une vue globale du terrain. En météorologie, les images satellites acquises à des temps différents sont recalées pour permettre une détection de changements. L'imagerie médicale bénéficie aussi d'outils de recalage pour associer des données IRM entre différents patients, et permettre ainsi une meilleure analyse des images.

L'équation 1.12 est la définition d'un recalage simple entre deux images, basé sur la minimisation de l'erreur quadratique entre les valeurs des pixels. Le recalage peut être en réalité plus complexe. Par exemple l'erreur quadratique minimisée ici peut être remplacée par un critère plus robuste. D'autre part, le recalage peut s'effectuer sur d'autres

caractéristiques de l'image que la valeur des pixels. Nous présentons ici un état de l'art général sur les méthodes existantes de recalage.

1.2.2 État de l'art du recalage d'images

Les méthodes de recalage se différencient principalement par les données de l'image considérées, le modèle de transformation utilisé, et la méthode d'optimisation permettant de trouver les paramètres optimaux de ce modèle.

1.2.2.1 Données considérées pour le recalage

Une première famille d'algorithmes en recalage d'images travaillent directement sur les valeurs des pixels de l'image, ou sur une transformation de celles-ci dans un autre espace couleur. Dans ce type de méthodes basées sur le recalage de surfaces, l'accent est plus mis sur la minimisation de l'erreur entre les deux images à recaler que sur les données à recaler.

L'autre famille d'algorithmes de recalage cherche à associer au mieux des primitives extraites dans chacune des deux images. Ces primitives sont par exemple des régions, des points, ou des lignes. Il s'agit donc d'un recalage de plus haut-niveau, en comparaison avec les méthodes de recalage de surfaces, car la correspondance entre images est ici réalisée à partir des primitives extraites, qui sont souvent pensées pour être robustes aux différentes perturbations possibles comme les différences d'illumination.

Régions Les approches basées sur l'extraction de régions de l'image fonctionnent à partir d'une segmentation des images en différentes régions de tailles adéquates. Ces régions sont ensuite représentées par leurs centres de gravité, qui est invariant aux rotations, changements d'échelle et cisaillement. De même, le bruit et les changements d'illumination influencent peu la position du centre de gravité. Par contre, la qualité de la segmentation est déterminante pour ce type de méthode. [GSP86] propose une solution à ce problème, avec un algorithme itératif réalisant la segmentation et le recalage des régions de manière conjointe. [AK03] extrait des régions invariantes aux changements d'échelle sous forme de cercles virtuels, et utilise une transformée de distance. Toujours dans l'optique de représenter les régions de manière invariante aux transformations, [TG04] décrit une méthode basée sur des voisinages invariants aux transformations affines, utilisant les coins de Harris [Nob88] et les contours passant par ces coins.

Lignes Les lignes sont aussi très populaires en recalage. Suivant les applications, elle peuvent correspondent aux contours des objets [LMM95] [DK97] [GSC98], aux bords de mer [SPM97] [MW87], aux routes [LKP92], ou à des structures anatomiques allongées [VB97]. La correspondance entre les segments est généralement exprimée par rapport aux paires d'extrémités ou aux points centraux. Les lignes sont détectées par des méthodes classiques de détection de contours comme le détecteur de Canny [Can86] ou de Sobel. Une comparaison des méthodes d'extraction de points de contours pour le recalage d'image a été réalisée par [MvdEV96b] [MvdEV96a].

Points De manière similaire aux lignes, plusieurs méthodes de recalage se basent sur la détection de points d'intérêt dans les images. Il peut s'agir de l'intersection de lignes [SKB82] [VZB98], d'intersection de routes [TJ89], de centres de régions, de points à forte variance ou dont la courbure présente une discontinuité, détectable par une transformée en ondelettes de Gabor [ZC93] [MSC96]. Le détecteur de coins de Harris [Nob88] extrait des points d'intérêt qui peuvent aussi permettre de trouver la correspondance géométrique entre deux vues d'une même scène. D'autres méthodes sont basées sur les extrema locaux de la transformée en ondelettes [FC97] [HLFK96], ou sur la détection des coins [BS97] [WSYR83] [HJP92].

1.2.2.2 Mesures de similarité

De manière générale, le recalage de deux images consiste à maximiser la similarité entre une première image et une autre image compensée en mouvement, en fonction des paramètres d'un modèle de transformation. La similarité entre les deux images peut être mesurée de plusieurs façons, en fonction du type de données à recaler. Nous différencions ici principalement les mesures de similarités pour les recalages de surfaces des mesures de similarités pour les recalages de primitives (régions, lignes, points).

Mesures pour le recalage de surfaces

Lorsque le recalage est effectué directement sur les valeurs des pixels, les mesures de similarité basées sur l'intercorrélation sont les plus couramment utilisées. L'intercorrélation normalisée $C(W_1, W_2)$ entre une fenêtre W_1 de l'image I_1 et une fenêtre de même taille W_2 de l'image I_2 est définie par :

$$C(W_1, W_2) = \frac{\sum_{\boldsymbol{k}} W_1(\boldsymbol{k}).W_2(\boldsymbol{k})}{\sqrt{\sum_{\boldsymbol{k}} W_1(\boldsymbol{k})^2}\sqrt{\sum_{\boldsymbol{k}} W_2(\boldsymbol{k})^2}} \tag{1.13}$$

Certaines méthodes utilisent l'intercorrélation pour l'estimation de transformations géométriques plus complexes que les seules translations [HF93]. Par exemple [Ber98] propose une méthode prenant en compte les transformations affines, et [Sim96] recale aussi les images qui diffèrent par la perspective et les imperfections de la lentille. Basé sur la même idée que l'intercorrélation, mais avec une implémentation plus rapide, l'algorithme SSDA [BH72] est une approche séquentielle qui mesure la somme des différences absolues entre chaque fenêtre, et teste cette somme par rapport à un seuil pour déterminer si les paramètres de la transformation sont corrects ou non. De même, l'erreur quadratique moyenne est utilisée par [WZ00] pour une estimation itérative de la déformation de perspective. Pour une meilleure robustesse aux variations d'intensité entre les deux images, certaines méthodes travaillent sur les contours des images. Ainsi, [HKR93] recale des images de contours en utilisant la distance de Hausdorff, ce qui semble plus performant

pour des images où la position des pixels est perturbée. [Pra74] applique un filtrage sur l'image avant de mesurer l'intercorrélation afin d'améliorer la robustesse au bruit. Enfin, [WS77] et [Anu70] calculent l'intercorrélation sur les images de contours.

Le domaine fréquentiel offre aussi certains avantages pour l'estimation des paramètres de mouvements 2D simples tels que les translations et les rotations. En effet ces transformations ont une expression simple dans le domaine de Fourier. Par contre, la nature de la transformation de Fourier limite les méthodes d'estimation aux modèles de mouvement global, tandis que le domaine spatial permet des modèles plus généraux, tels que les homographies. Aussi, un avantage en faveur des méthodes travaillant dans le domaine spatial est qu'elle peuvent être appliquées tant à l'image entière qu'à un sous-ensemble de l'image [CZ03]. Les translations dans le domaine spatial correspondent à un décalage de phase entre les deux images, qui peut être trouvé par corrélation. D'autre part, en appliquant une transformation log-polaire de la magnitude du spectre de fréquence, les rotations et changements d'échelle correspondent à des décalages horizontaux et verticaux, et peuvent être estimés par une méthode de corrélation de phase. [RC96] décrit une telle approche pour l'estimation d'un mouvement planaire, et appliquant un filtre renforçant les hautes fréquences de l'image pour une meilleure estimation. [MBM97] décrit aussi une méthode similaire dans l'espace de Fourier. [SOCM01], [VSV03] et [VSV05] utilisent eux aussi une technique de corrélation de phase pour l'estimation d'un mouvement planaire, en travaillant sur la partie basse-fréquence des images pour réduire les erreurs dues à l'aliasing.

Un autre exemple concerne le recalage de deux images qui appartiennent à des modalités différentes, lorsque le but est par exemple de recaler des données IR à des données UV. Dans ce cas, l'intercorrélation n'est pas une mesure pertinente et on préférera un algorithme qui maximise l'information mutuelle. [VWI95] ont présenté une méthode basée sur l'information mutuelle pour le recalage d'images de résonance magnétique. La maximisation est réalisée par descente de gradient. D'autres algorithmes d'optimisation ont été étudiés, comme la méthode Jeeves [TU96] ou l'algorithme de Marquardt-Levenberg [TU98].

Mesures pour le recalage de primitives

Les mesures de similarité utilisées pour le recalage de primitives isolées de l'image sont souvent très différentes des mesures pour le recalage de surface. Elles sont aussi plus rapide à calculer, car elles portent sur un ensemble de données plus restreint que l'image entière.

Le recalage de chanfrein, introduit par [BTBW77] [Bor88] cherche à minimiser une distance globale entre les contours des deux images à recaler. La mesure de similarité consiste à calculer la transformée de distance sur une des images de contours, et à en déduire la distance moyenne aux contours de l'autre image compensée en mouvement.

Le recalage peut aussi être vu comme un problème d'association de paires de primitives entre les deux images, à partir d'une description de ces primitives. La description la plus simple considère la valeur des pixels dans le voisinage de chaque primitive extraite [Leh98]. La correspondance entre primitives est alors estimée à partir de l'intercorrélation entre ces voisinages. La méthode proposée par [ZC93] utilise les coefficients de corrélation, tandis

que [ZFS02] se base sur l'information mutuelle. Les descriptions des primitives sont parfois plus complexes que la simple valeur des pixels du voisinage. Par exemple [Mur92] associe à chaque point la distribution spatiale des primitives voisines, tandis que [ZK99] décrit chaque point par les angles entre les lignes qui s'intersectent en ce point. Les descripteurs SIFT [Low99] sont populaires pour obtenir une description des points d'intérêt invariante au facteur d'échelle.

1.2.2.3 Modèles de transformation

Le modèle de mouvement, représenté par la fonction $f_{\boldsymbol{m}} : \mathcal{D} \longrightarrow \mathcal{D}$, peut être plus ou moins complexe, avec plus ou moins de degrés de liberté. Le modèle le plus simple généralement utilisé est le modèle de translation, que l'on peut retrouver dans d'autres techniques d'analyse d'images comme le blockmatching. Il a l'avantage de ne posséder que 2 paramètres, qui sont les coordonnées du vecteur de translation. Ce modèle suppose bien évidemment que le mouvement apparent peut être assimilé à une translation, ce qui est le cas le déplacement de la caméra est suffisamment faible. On peut étendre ce type de déplacement à un modèle affine plus complet [IP91], comprenant aussi les rotations, changements d'échelle et cisaillements. Ce modèle comprend 6 paramètres, et la fonction $f_{\boldsymbol{m}}$ est la suivante :

$$f_{\boldsymbol{m}}(\boldsymbol{k}) = \begin{bmatrix} m_1 & m_2 \\ m_3 & m_4 \end{bmatrix} .\boldsymbol{k} + \begin{bmatrix} m_5 \\ m_6 \end{bmatrix} \tag{1.14}$$

avec m_1, m_2, m_3, m_4 les paramètres correspondant aux rotations, changements d'échelle et cisaillements, et m_5, m_6 les paramètres du vecteur de translation. Il existe aussi des modèles non linéaires pour le recalage élastique de données, introduit par [BK89]. Ce type de recalages non linéaires convient surtout pour des domaines d'application tels que l'imagerie médicale, lorsque des acquisitions IRM, sur des sujets différents, doivent être recalés.

1.2.2.4 Estimation des paramètres de la transformation

Enfin, la méthode de minimisation elle-même peut changer selon les algorithmes de recalage. La recherche exhaustive du vecteur $\boldsymbol{m}$ optimal dans l'espace $\mathcal{M}$ des paramètres de la transformation n'est pas toujours possible, pour des raisons de temps de calcul. La minimisation classique par descente de gradient explore l'espace des paramètres dans la direction au gradient de l'erreur, et peut être utilisée lorsque le risque de minimum local est réduit. Cela impose une bonne initialisation des paramètres de départ, ce qui n'est pas toujours possible. Les méthodes multirésolution permettent aussi de ne pas parcourir tout l'espace des paramètres. [BAHH92] a par exemple introduit une méthode hiérarchique pour estimer le mouvement dans une structure de données multirésolution.

Pour les problèmes multimodaux où le risque d'atteindre un maximum local s'avère trop élevé, les méthodes d'optimisation basées sur l'échantillonnage aléatoire sont plus adaptées. Elle permettent d'éviter un parcours exhaustif des paramètres possible de la transformation géométrique, tout en réduisant le risque de convergence de le mesure de

similarité vers un extremum local. [FWKP05] utilise un filtrage particulaire pour recaler des données 2D à des données 3D.

1.2.3 Conclusion sur les méthodes existantes de recalage

Nous verrons dans la suite que les images dont nous souhaitons estimer le mouvement relatif apparent sont des images de contours, pour des raisons de robustesse aux perturbations de la scène. Elles contiennent principalement des hautes fréquences, ce qui nous empêche d'utiliser des méthodes de type multirésolution ou descente de gradient pour l'estimation des paramètres de transformation. Ces méthodes s'appuient en effet sur l'information basse-fréquence dans l'image. Les points de contours que nous allons extraire peuvent être vus comme des primitives de l'image. Cela nous incite à privilégier les méthodes basées sur la mise en correspondance de primitives. Parmi les méthodes existantes, nous nous intéressons principalement à celle proposée par [Bor88], où la mise en correspondance est réalisée au travers d'une transformée de distance. Cette méthode est en effet peu coûteuse en temps de calcul, et peut être adaptée à des situations où la mesure de similarité doit être robuste. Néanmoins, la méthode plus naturelle de recalage par maximisation de l'intercorrélation normalisée (équation 1.13) entre les deux cartes de contours sera d'abord étudiée, parce que cette mesure est par nature plus fiable qu'une mesure basée sur la transformée de distance, et parce qu'elle fonctionne sur des données en niveau de gris.

Chapitre 2

Mesures de qualité du système d'acquisition

Sommaire

2.1 Mesure de la position d'une caméra 33
2.1.1 Points de contours stables 33
2.1.1.1 Détection des points de contour 33
2.1.1.2 Extraction des points de contours stables 36
2.1.2 Comparaisons entre vues 39
2.1.2.1 Estimation de mouvement apparent par corrélation . . 40
2.1.2.2 Estimation rapide du mouvement par une carte de distance 41
2.1.2.3 Correction des distortions radiales de l'objectif 46
2.2 Mesure de la netteté de l'image 48
2.2.1 Mesures efficaces de la taille des contours 48
2.2.1.1 Modèle de profil de la norme du gradient 48
2.2.1.2 Approximation de la largeur moyenne des contours . . . 51
2.2.1.3 Invariance au contenu des mesures de netteté 51
2.3 Obstruction du champ de vision par un objet 52
2.3.1 Effets possibles d'une obstruction 54
2.3.2 Visibilité des éléments fixes de la scène 55
2.3.2.1 Proportion visible des contours stables 55
2.3.2.2 Amélioration de la détection des obstructions partielles 57
2.4 Conclusions sur les mesures de qualité du système d'acquisition 57

Le système de vidéosurveillance doit être conçu de manière à limiter ou faciliter au plus les interventions humaines, qui peuvent devenir lourdes lorsque le nombre de véhicules à installer et entretenir devient important. L'analyse des images des caméras de surveillance est un outil qui permet de tendre vers ce but, en extrayant les informations nécessaires

pour que le système puisse connaître son état, et réagir en conséquence. Les informations à extraire concernent directement l'état du système d'acquisition, sans tenir compte précisément du contenu de la scène. Dans cette étude nous nous sommes limités à trois caractéristiques majeures de la caméra, qui nous semblent décrire l'état du système de manière relativement complète. Il s'agit du champ de vision de la caméra, de la netteté de ses images renvoyées, et de l'occultation ou non de la scène par un objet gênant une bonne exploitation des images. Pour chacune de ces caractéristiques, il existe un état idéal, dans lequel la caméra devrait se situer. Concrètement, la caméra doit être *positionnée correctement*, *bien focalisée* et le champ de vision ne doit *pas être occulté* par un objet. L'analyse des images va permettre de mesurer la qualité du système d'acquisition à travers ces trois caractéristiques, de les comparer aux états idéaux et d'en informer le système afin qu'il réagisse de manière appropriée.

Dans ce chapitre nous présentons des algorithmes permettant d'extraire des mesures pour chacune des trois caractéristiques auxquelles nous nous intéressons. Ces mesures doivent répondre à deux propriétés essentielles aux applications que nous envisagerons par la suite :

1. Elle doivent être corrélées à la caractéristique considérée, afin de refléter un changement de l'état du système d'acquisition, pour permettre sa détection.

2. Elle doivent être stables dans le temps, en l'absence de changement de l'état du système d'acquisition, et en dépit des variations des images.

Pour caractériser la position de la caméra, nous introduisons ici l'extraction des points de contours stables de la vue, correspondant aux bords des éléments fixes de l'intérieur du véhicule. Ces contours ont la propriété d'être indépendants aux fortes variations en mouvement et en illumination de la scène, tout en représentant efficacement le champ de vision de la caméra. La mesure associée au champ de vision sera ensuite présentée. Il s'agit d'une mesure de distance entre les contours stables de la vue et des contours stables de référence, extraits pour une position de caméra considérée comme idéale.

Pour mesurer la qualité de mise au point de la caméra, deux mesures de netteté sont proposées, basées sur l'estimation de la largeur des contours. Contrairement aux mesures existantes, adaptées à des applications comme l'autofocus ou l'estimation de profondeur (*depth from defocus*), les mesures proposées sont conçues pour être stables par rapport au contenu de la scène et à son illumination. L'une des deux mesures est pensée comme une approximation de la première, afin de caractériser la netteté par un calcul rapide.

Enfin, l'occultation du champ de vision par un objet gênant est une caractéristique plus délicate à mesurer, car elle ne concerne pas directement le système d'acquisition, et ne peut pas être représentée par un modèle physique clair. Après une définition de l'occultation de la scène d'un point de vue image, nous proposons une mesure de cette caractéristique basée sur la présence ou l'absence des points de contours stables, introduits en premier lieu pour la mesure de la position de la caméra.

2.1 Mesure de la position d'une caméra

Certaines applications envisagées, que nous présentons plus loin, requièrent que le système possède une information relative à la position de la caméra. Il est difficile de juger de façon automatique si une caméra de surveillance est placée correctement ou non, mais il est déjà beaucoup plus facilement envisageable de comparer une position de caméra à une autre position idéale, dite de référence. Pour réaliser cela, il nous faut définir et extraire une caractéristique de la vidéo qui rend compte de la position, tout en étant invariante aux différents changements temporels de la scène, tel que les variations d'illumination et les mouvements de passagers et du paysage extérieur. On souhaite effectivement extraire une information qui puisse discriminer des positions de caméra différentes, et qui est stable dans le temps.

2.1.1 Points de contours stables

La position des points de contour dans une image est une information qui peut être extraite par des filtres classiques comme Sobel ou Canny [Can86]. Les contours ont une propriété qui nous intéresse tout particulièrement : leur position est invariante aux variations d'illumination dans l'image. Effectivement, il s'agit d'une information dépendant de la structure de l'image, c'est-à-dire de la forme des objets qui la compose plus que de leurs valeurs photométriques. Ainsi, les différences entre deux cartes de points de contour extraites d'une même caméra à deux instants différents ne vont concerner que certains objets de la scène, qui seront apparus, disparus, ou qui auront été en mouvement entre ces deux instants. En réalité, comme les algorithmes classiques de détection de contours se basent sur la détection des changements spatiaux forts, une carte de points de contour contiendra aussi certains éléments de la scène liés à l'illumination, notamment les ombres portées. En somme, les contours d'une image provenant de nos caméra incluent :

- Les contours des éléments fixes à l'intérieur du véhicule, tels que les sièges, le rebord des vitres, les barres métalliques, etc.
- Les contours des objets mobiles à l'intérieur du véhicule, notamment les passagers
- Les contours du paysage extérieur vu à travers les vitres
- Les contours créés par les ombres portées

Afin de caractériser la position de la caméra par une information stable en fonction du temps, on souhaite ne considérer que les contours de la scène dont la position ne change pas au cours du temps. Ces contours correspondent donc aux contours des éléments fixes à l'intérieur du véhicule. On parlera alors des *points de contours stables*.

2.1.1.1 Détection des points de contour

Avant d'extraire les contours stables de la scène, il est nécessaire de détecter les points de contours pour chaque image de la séquence. Un point de contour est classiquement défini comme un point de l'image où la norme du gradient spatial est importante, supérieure à un seuil défini selon un critère donné. De nombreux algorithmes existent pour la détection

de contours dans l'image. Ils sont plus ou moins efficaces en présence de bruit, et ont des complexités variées. Les détecteurs de contours intègrent généralement un filtre de lissage pour réduire l'effet du bruit, puis appliquent une dérivée spatiale pour faire ressortir les zones à fort contraste local.

Détecteur de Canny Le détecteur de Canny consiste en un lissage de l'image en niveau de gris par une convolution Gaussienne, suivie d'un opérateur calculant la dérivée spatiale au premier ordre. L'image contient alors des crêtes autour des points de contours à détecter. L'algorithme suit ensuite le haut de ces crêtes, en fixant à zéro les pixels qui ne sont pas maximum localement. Le parcours des crêtes est contrôlé par deux seuils T_1 et T_2 ($T_1 > T_2$), avec un fonctionnement par hystérésis. Le suivi commence seulement sur un point de la crête supérieur à T_1, et continue dans les deux directions possibles à partir de ce point jusqu'à ce que la hauteur soit inférieur à T_2. De cette façon, les contours détectés sont moins susceptibles d'être coupés en plusieurs petits segments à cause du bruit.

Détecteur de Sobel D'autres détecteurs de contours moins complexes, qui ne sont pas basés sur un suivi des crêtes, apportent aussi des résultats satisfaisant. Le filtre de Sobel calcule le gradient spatial de l'image lissée par une convolution Gaussienne avec seulement deux opérateurs matriciels (horizontal et vertical). Pour une taille de fenêtre de convolution 3×3, les opérateurs horizontal H et vertical V sont :

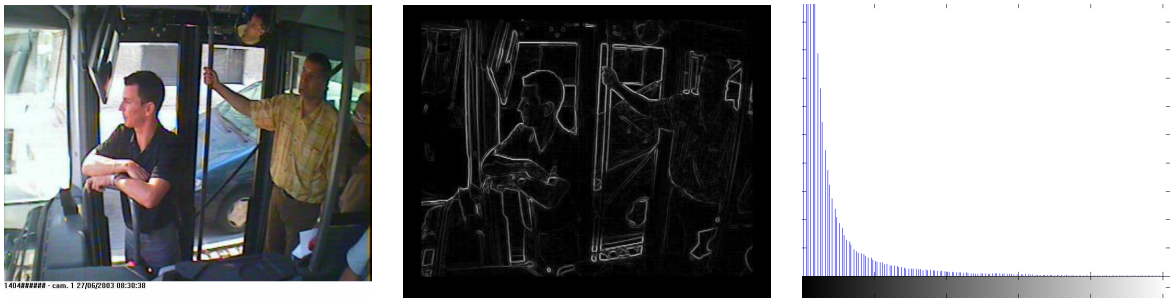
$$H = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad V = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \tag{2.1}$$

Un avantage algorithmique du détecteur de Sobel est que l'on peut séparer chaque opérateur 2D par deux convolutions 1D consécutives. Par exemple, H s'écrit comme le produit de H_1 et H_2 :

$$\begin{aligned} H &= H_1.H_2 \\ H &= \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix} . \begin{bmatrix} 1 & 2 & 1 \end{bmatrix} \end{aligned} \tag{2.2}$$

Les faibles ressources machine imposées par le système embarqué nous incitent à privilégier le détecteur de Sobel, certes plus simple et moins performant, plutôt que le détecteur de Canny. En réalité, l'extraction des contours stables présentée dans la suite met en œuvre un filtrage temporel qui va permettre de réduire les imperfections du filtre de Sobel face aux images bruitées.

Ces convolutions résultent en un ensemble de vecteurs de gradient, dont on calcule les normes (figure 2.1(b)). La détection des contours à proprement parler nécessite de seuiller le gradient de l'image par une valeur que l'on doit définir.



(a) (b) (c)

FIG. 2.1 – Norme du gradient calculé par le filtre de Sobel et histogramme des normes

Choix automatique du seuil de détection Les images issues des caméras de surveillance étudiées sont suffisamment variées en terme d'illumination pour que l'on ait besoin de déterminer automatiquement le seuil de détection des points de contours. En effet une image sombre présentera des contours au gradient moins fort qu'une image fortement éclairée, et un seuil trop haut ne détectera pas les contours de l'image sombre. De même, une image éclairée aura un gradient plus fort dans les zones ne comportant pas de contours, et un seuil trop faible provoquera beaucoup de fausses détections. La luminosité des images peut changer très vite, d'une image à l'autre, dans des situations telles que le passage sous un tunnel. De plus, le contrôle automatique de gain que possède la caméra ne parvient à réduire les différences de luminosité que jusqu'à un certain degré. La recherche d'un seuil T propre à chaque image est donc nécessaire.

Pour déterminer T, nous faisons l'hypothèse que les normes des vecteurs de gradient de l'image lissée sont des échantillons positifs issus d'une distribution gaussienne centrée $\mathcal{N}(0, \sigma^2)$, pour l'ensemble des points de l'image qui ne sont pas des points de contours. Cette hypothèse parait raisonnable d'après l'histogramme des normes, figure 2.1(c). Les normes des vecteurs de gradient aux points de contours sont quant-à elles supposées fortes et en dehors de la distribution. On souhaite tout d'abord estimer la variance de cette distribution des normes. Cette estimation doit être réalisée de manière robuste aux points de contours. Pour cela, seule une certaine proportion n_f des normes les plus faibles est donc considérée pour le calcul de la variance σ^2.

L'ensemble des $n_f.N$ normes les plus faibles (N étant le nombre de points de l'image) est obtenu en construisant l'histogramme h des normes, ce qui évite d'avoir à trier toutes les normes. La variance est alors calculée à partir de l'histogramme :

$$\sigma^2 = \sum_{i=0}^{i_{max}} h(i).i^2 \tag{2.3}$$

avec i_{max} le plus petit entier positif tel que

$$\sum_{i=0}^{i_{max}} h(i) > n_f.N \tag{2.4}$$



(a) (b) (c)

FIG. 2.2 – Détection des contours dans une image. (a) : image originale (b) : détection par le filtre de Canny (c) : détection par le filtre de Sobel

Lorsque la variance des normes des points qui ne sont pas contours est estimée, on détermine le seuil T décidant si une norme est assez forte pour être celle d'un point de contour, comme proportionnel à l'écart-type : $T = 3\sigma$. Si l'hypothèse de gaussianité des normes est respectée, les normes en dessous du seuil représentent 99% des normes ne correspondant pas à un contour. Les normes du gradient ne sont bien entendu pas distribuées de manière gaussienne dans la réalité, mais l'hypothèse semble suffisante pour que l'on obtienne des résultats très satisfaisants pour la grande majorité des images.

Détection des contours à partir de la norme du gradient On suppose que les points de contours de l'image sont ceux qui répondent au deux conditions suivantes :

1. La norme du gradient doit être supérieure au seuil T.
2. La norme du gradient doit être maximale localement, dans au moins une des deux directions, horizontale ou verticale.

La seconde condition permet d'obtenir des lignes de contours fines. En effet, lorsque l'image est floue, les contours sont étalés et le gradient à une norme élevée sur plusieurs pixels contigus.

Comparaisons des détecteurs de Canny et Sobel La figure 2.2 présentent les résultats de détection de contours pour une même image, avec le détecteur de Canny et le détecteur de Sobel. Le principal avantage du détecteur de Canny est qu'il produit des contours qui ne sont pas discontinus, grâce à son seuillage hystérésis. Sa complexité de calcul plus importante nous fait privilégier tout de même le détecteur de Sobel, qui est de toute façon suffisamment performant pour l'extraction des contours stables présentée dans la suite.

2.1.1.2 Extraction des points de contours stables

L'extraction des points de contours stables est réalisée par un filtrage temporel des contours au cours de la séquence vidéo. Les points de contour de l'intérieur du véhicule

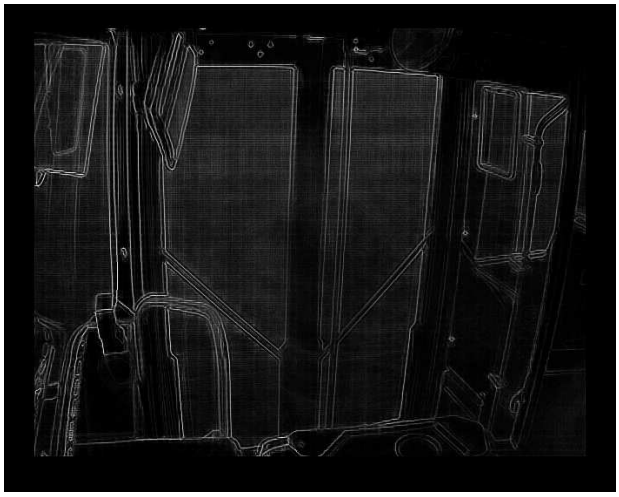


FIG. 2.3 – Carte des contours stables

ont en effet un comportement temporel particulier par rapport aux autres contours de la scène. Ils apparaissent toujours à la même position dans l'image lorsqu'ils sont visibles, et ils sont parfois cachés temporairement par des objets de la scène. Il est possible d'extraire ces points de contours en moyennant temporellement les cartes de contours.

La figure 2.3 illustre le résultat d'une moyenne temporelle des cartes de contours sur trois heures de vidéo, pour la caméra dont l'image 2.2(a) est issue. Les contours du paysage extérieur ont bien disparu, ainsi que les contours des passagers et des ombres.

La détection des contours (seuillage et recherche des maxima locaux de la norme du gradient) pour chaque image est primordiale. Elle accorde une importance égale à tous les contours de l'image, quelque soit leur saillance. Ainsi la carte des contours stables apporte bien une information sur la fréquence d'apparition des contours, indépendante de la norme du gradient en ces points de contour.

La carte des contours stables 2.3 n'est quant-à elle pas binarisée afin de ne pas perdre trop d'information. Elle caractérise la position de la caméra pour le temps pendant lequel la moyenne est calculée.

Considérations sur l'implémentation de l'extraction des contours stables L'extraction des contours stables est un algorithme bas niveau, dont le résultat va être utilisé par des traitements ultérieurs. Ces traitements ont besoin d'une carte de contours stables qui reflète la position de la caméra à l'instant courant. La carte doit donc être mise à jour le plus régulièrement possible, et doit être calculée sur une durée assez courte pour ne pas être influencée par les positions précédentes de la caméra (dans le cas où la caméra aurait été déplacée). L'intégration des contours de chaque image sur 3 heures (durée totale d'une de nos séquences vidéo de test), comme présenté figure 2.3 produit de bons résultats, mais cette durée est trop longue pour qu'un changement de position de caméra soit pris en compte rapidement. En contrepartie, plus le temps d'intégration des cartes de contours de chaque image est long, meilleure est la carte de contours stables résultante. Un compromis

entre temps d'intégration et régularité de la mise à jour doit être fait.

Pour obtenir une mise à jour régulière des contours stables, il nous faut calculer une moyenne glissante, sur une fenêtre temporelle dont la durée reste à définir. Les ressources mémoire étant limitées, il est évidemment exclu de mémoriser les cartes de contours pour toutes les images de la fenêtre temporelle.

Une première façon de calculer la carte de contours stables sur une fenêtre temporelle F glissante consiste à diviser F en m fenêtres consécutives plus courtes $F_1, F_2, \ldots, F_m$. Les cartes de contours stables S_{F_i} sont calculées sur chaque fenêtre F_i, et mémorisée, ce qui ne requiert qu'un espace mémoire équivalent à m cartes. La carte de contours stables finale S_F est obtenue comme la moyenne des S_{F_i}. On réalise une moyenne glissante en supprimant la carte la plus ancienne S_{F_1} et en considérant la fenêtre temporelle suivante F_{m+1}, lorsque le calcul sur F_m est terminé. Il s'agit ici d'une moyenne glissante par blocs, qui donne de bons résultats en terme de régularité de la mise à jour, tout en occupant un espace mémoire relativement réduit.

Une seconde façon de réaliser une moyenne glissante, plus élégante que la première et procurant des résultats similaires, consiste à simuler une moyenne pondérée en fonction de l'*âge* de chaque image. Plus une image est ancienne par rapport à l'instant courant, moins elle contribuera à la carte de contours stables. Pour une fenêtre temporelle de taille T (c'est à dire contenant T images), la carte de contours stables S_{t-1} est mise à jour avec la carte de contours C_t au temps t par :

$$S_t = \frac{1}{T}\left((T-1).S_{t-1} + C_t\right) \tag{2.5}$$

La contribution d'un point de contour apparu au temps t_1 dans la carte des contours stables au temps t suit une loi en

$$c(t-t_1) = \frac{1}{T}\left(\frac{T-1}{T}\right)^{(t-t_1)} \tag{2.6}$$

Cela signifie que la valeur en une position donnée dans la carte de contours stables au temps t est la somme des $c(t-t_1)$ pour tout les instants t_1 où un point de contour est présent en cette position. L'évolution de la contribution c d'un point de contour à la carte des contours stables en fonction de sa date d'apparition est illustrée figure 2.4. On remarque d'ailleurs que

$$\sum_{t'=0}^{\infty} c(t') = 1 \tag{2.7}$$

ce qui correspond à la valeur maximale de la carte des contours stables, lorsqu'un point de l'image est un point de contour toujours visible.

Ce type de moyenne glissante présente deux avantages par rapport à la méthode précédente :

– Contrairement à la première méthode, il n'y a pas de discontinuité temporelle dans l'évolution de la carte de contours stables. Des discontinuités apparaissent avec la

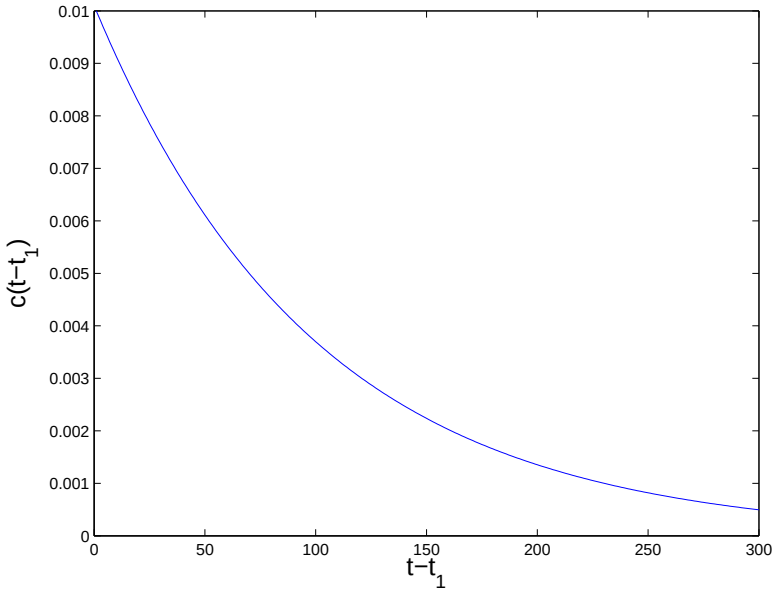


FIG. 2.4 – Contribution d'un point de contour en fonction de la date de son apparition, pour une fenêtre temporelle de taille $T = 100$

première méthode lorsqu'on passe d'une fenêtre temporelle à la suivante, car cela annule la contribution des points de la fenêtre F_1 en un seul instant.
– Cette méthode requiert moins de mémoire que la première. Seulement une carte de contours stables doit être mémorisée.

La contribution des points de contour à la carte des contours stables est par contre plus difficile à caractériser. En théorie, d'après l'équation 2.6, un point de contour contribue indéfiniment aux contours stables. En pratique, la contribution du point devient négligeable après un temps supérieur à $3.T$.

2.1.2 Comparaisons entre vues

La carte de contours stables, extraite des données vidéo par la méthode qui vient d'être définie, est une caractéristique qui rend compte de la position de la caméra en se basant sur des éléments statiques de la scène. En plus d'être une caractéristique qui diffère pour des positions de caméra différentes, elle présente l'avantage de permettre la comparaison entre deux positions différentes d'une même caméra de manière relativement naturelle. Plus précisément, l'extraction des contours stables peut être interprétée comme un filtrage de toutes les perturbations en mouvement et en illumination qui interviennent dans notre application. Ce filtrage conserve tout de même la structure générale de la scène, et il est par conséquent possible de comparer deux cartes de contours stables comme l'on comparerait deux images. En particulier, dans notre contexte nous nous intéressons au bon placement de la caméra, qui est à une position fixe dans le véhicule mais qui doit être

orientée correctement pour délivrer des images exploitables. La comparaison entre deux vues d'une même caméra correspond à un problème de recalage entre ces deux vues, qui est un problème classique de traitement d'image lorsque ces deux vues contiennent en partie les même objets. Un état de l'art des méthodes de recalage à été établi dans le chapitre 1. Nous présentons ici les méthodes développées pour comparer deux vues provenant d'une même caméra à partir des cartes de contours stables.

2.1.2.1 Estimation de mouvement apparent par corrélation

Une première approche pour l'estimation du mouvement apparent entre deux cartes de contours stables consiste à parcourir l'espace des paramètres du modèle de mouvement de façon exhaustive (en se limitant toutefois aux valeurs probables des paramètres), afin de minimiser l'erreur quadratique moyenne entre la seconde carte de contours stables et la première carte compensée en mouvement (équation 1.12).

Le temps de calcul de cette méthode est son inconvénient majeur. Nous limitons la recherche aux seules translations entre les deux cartes, ce qui est raisonnable pour notre application, d'après les mouvements possibles de la caméra. En effet la caméra peut être orientée selon deux axes, horizontal et vertical, ce qui provoque principalement un déplacement en translation de la scène. Dans un autre contexte ou le mouvement apparent peut aussi contenir des rotations ou des changements d'échelles, l'espace des paramètres serait trop vaste pour un parcours exhaustif.

D'autre part, l'espace de l'image étant limité, un déplacement de la vue provoque une disparition d'une partie de la scène et une apparition d'une autre partie. La recherche du vecteur de translation consiste donc plus précisément à minimiser l'erreur quadratique entre deux fenêtres 2D à l'intérieur de la carte de contours. Concrètement, on considère 4 fenêtres rectangulaires, chacune placée dans un des coins de la première carte de contours, puis on cherche la fenêtre de même taille dans la seconde carte de contours, produisant l'erreur quadratique minimale. Cela permet de prendre en compte un espace de paramètres suffisamment large pour prendre en compte les translations importantes, quelque soit leur direction. La taille de la fenêtre correspond en pratique à un quart de la taille de l'image.

La minimisation de l'erreur quadratique moyenne peut être remplacée par une maximisation de la corrélation entre les fenêtres, qui est une mesure équivalente mais plus rapide en temps de calcul.

Validation de l'estimation du mouvement apparent Bien que les seuls mouvements de la caméra autorisés par le matériel produisent des translations de l'image, il est fort probable que le mouvement apparent estimé par le modèle de translation soit très loin de la réalité. Cela se produit lorsque le déplacement de la caméra est trop important, supprimant la zone en commun dans les deux cartes de contours stables. Il est aussi possible que la caméra soit détachée de son socle, suite à un acte de vandalisme par exemple, auquel cas les mouvements apparents possibles ne sont plus limités aux seules translations. Il paraît donc nécessaire de pouvoir valider ou non le résultat de l'estimation de mouvement, de façon automatique.

On utilise pour cela une mesure de corrélation γ normalisée entre les deux cartes de contours recalées, dans la fenêtre d'analyse considérée pendant l'estimation. En dénotant par W_1 et W_2 les fenêtres recalées des cartes de contours C_1 et C_2 respectivement :

$$\gamma(W_1, W_2) = \frac{\sum_{y=1}^{N_y}\sum_{x=1}^{N_x} W_1(x,y).W_2(x,y)}{\sqrt{\sum_{y=1}^{N_y}\sum_{x=1}^{N_x} W_1(x,y)^2 \sum_{y=1}^{N_y}\sum_{x=1}^{N_x} W_2(x,y)^2}} \tag{2.8}$$

avec $N_x \times N_y$ les dimensions de la fenêtre d'analyse.

La corrélation normalisée est une mesure comprise entre 0 et 1, qui permet de caractériser la similarité entre les deux fenêtres. Si $\gamma(W_1, W_2)$ a une valeur trop faible, cela signifie que l'estimation de mouvement a échouée. Elle reflète la qualité de l'estimation. On décide si le mouvement estimé est valide ou non en seuillant cette corrélation à une valeur déterminée empiriquement à 0.8.

Détection de mouvement par corrélation On compare généralement une carte de contours d'une caméra à un temps donné, avec une autre carte de contours de la même caméra calculée à un temps de référence. En réalité, les caméras du véhicule sont normalement fixes. Les seules occasions où un mouvement peut apparaître sont :

- lors de l'installation de la caméra, lorsqu'il s'agit d'orienter correctement la caméra
- lorsque la caméra se dérègle pendant son fonctionnement, soit parce qu'elle était mal fixée, soit parce qu'elle a subi un acte de vandalisme.

De ce fait, la caméra conserve une position fixe pendant la majeure partie de son temps de fonctionnement, et le mouvement apparent estimé est donc très souvent identique. Le calcul d'estimation de mouvement est réalisé en permanence, afin qu'un déplacement non prévu de la caméra puisse être détecté. Comme il s'agit d'un calcul lourd, on souhaite modifier l'algorithme afin d'alléger les ressources machines qu'il demande. Pour cela, l'estimation de mouvement n'est réalisée que lorsqu'on pense qu'un mouvement a eu lieu, par rapport à la dernière estimation de mouvement qui a été réalisée. La corrélation normalisée, équation 2.8, donne une indication sur la validité du dernier déplacement estimé. Si cette corrélation est trop faible, c'est à dire en dessous du seuil de 0.8, on considère qu'un mouvement a pu se produire depuis la dernière estimation. Dans le cas contraire, on conserve le dernier déplacement estimé. Le calcul de la corrélation normalisée est beaucoup plus rapide que l'estimation de mouvement.

2.1.2.2 Estimation rapide du mouvement par une carte de distance

La lenteur de l'algorithme d'estimation de mouvement basé sur un parcours exhaustif de l'espace des paramètres nous incite à envisager une méthode plus rapide, quitte à ce qu'elle soit moins précise. Une carte de contours stables peut être interprétée comme un

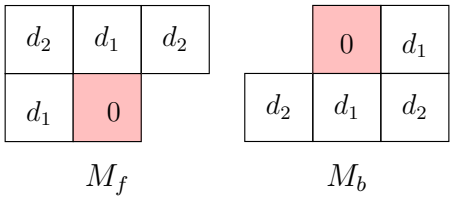


FIG. 2.5 – Masques de chanfrein

ensemble de points d'intérêt de l'image. La *distance* entre deux cartes de contours stables peut être calculée, par l'intermédiaire d'une carte de distance.

Détection des points de contours stables Il est tout d'abord nécessaire de binariser les cartes de contours stables, pour obtenir un ensemble de points d'intérêt. En effet, une carte de contours est un ensemble de valeurs réelles, relatif à la fréquence d'apparition d'un point de contour en chaque position de l'image. Comme pour la détection des points de contours à partir du gradient, présentée précédemment, on réalise un seuillage de la carte des contours stables. Le seuil est déterminé automatiquement pour chaque image, en modélisant la distribution des valeurs de la carte qui ne sont des contours stables par une gaussienne centrée. Lorsque la variance σ^2 de ces valeurs est estimée, on choisit comme seuil $3.\sigma$. Seuls les points de la carte de contours qui ont une valeur supérieure à ce seuil, et qui sont des maxima locaux, sont considérés comme des points de contours stable.

Calcul de la carte de distances La carte de distance est une carte de mêmes dimensions que la carte de contours stables, pour laquelle la valeur en un point est la distance au point de contour le plus proche.

Le calcul de la carte de distance est réalisable en un temps linéaire en fonction du nombre de pixels. On utilise pour cela une transformation de chanfrein, qui fait l'hypothèse qu'il est possible de déduire la valeur de la distance en un pixel à partir de la valeur de la distance en ses voisins. Cette hypothèse est vraie pour les distances d pour lesquelles l'affirmation suivante est vraie [RK76] :

$$\forall\ \boldsymbol{p}, \boldsymbol{q} \quad \textit{tels que} \quad d(\boldsymbol{p}, \boldsymbol{q}) \leq 2, \quad \exists\ \boldsymbol{r} \neq \boldsymbol{p}, \boldsymbol{q} \quad \textit{tel que} \quad d(\boldsymbol{p}, \boldsymbol{q}) = d(\boldsymbol{p}, \boldsymbol{r}) + d(\boldsymbol{r}, \boldsymbol{q}) \tag{2.9}$$

La distance euclidienne ne répond pas à cette propriété. Par contre, certaines approximations ont cette propriété, comme la distance *Manhattan* :

$$D_4((x_1, y_1), (x_2, y_2)) = |x_1 - x_2| + |y_1 - y_2| \tag{2.10}$$

ou la distance de Chebyshev (aussi connue sous le nom de distance *Chessboard*) :

$$D_8((x_1, y_1), (x_2, y_2)) = \max(|x_1 - x_2|, |y_1 - y_2|) \tag{2.11}$$

Les transformations de distance de chanfrein consistent en deux parcours consécutifs de la carte des distances par les masques M_f et M_b de la figure 2.5. La carte des distances

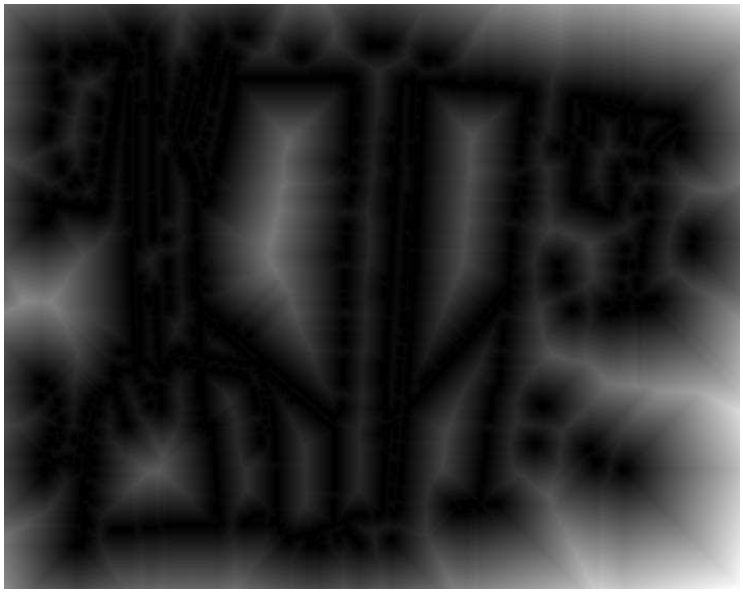


FIG. 2.6 – Carte de distances aux contours stables

est tout d'abord initialisée à 0 aux points de contours stables et à ∞ aux autres points. Le premier parcours est réalisé avec le masque M_f de gauche à droite puis de haut en bas, tandis que le deuxième parcours est réalisé avec le masque M_b dans le sens inverse. A chaque passage du masque sur un pixel, les distances d_1 et d_2 sont ajoutées aux valeurs courantes de la carte de distances et on choisit pour la nouvelle distance du centre du masque (à la position du zéro) le minimum de ces cinq sommes. Les valeurs d_1 et d_2 sont choisies de manière à minimiser la différence entre la carte de distance obtenue par les masques de chanfrein et celle qui serait obtenue avec une mesure de distance euclidienne [Bor84] :

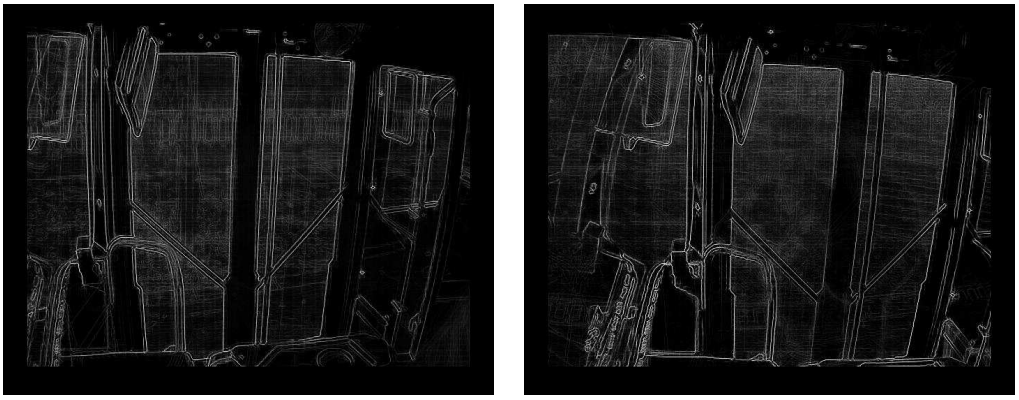
$$\begin{cases} d_1 &= 1 \\ d_2 &= \dfrac{1}{\sqrt{2}} + \sqrt{\sqrt{2}-1} \;\approx\; 1.351 \end{cases} \tag{2.12}$$

Pour accélérer le calcul, on utilisera plutôt les valeurs entières $d_1 = 3$ et $d_2 = 4$. Les distances obtenues devront être divisées par 3 pour représenter des distances dont l'unité est le pixel.

La figure 2.6 représente la carte de distance obtenue par la transformation de chanfrein sur la carte de contours stables de la figure 2.3.

Calcul de la distance entre deux vues La carte de distances obtenue sur une position de la caméra permet de mesurer la distance entre cette vue et une autre vue de la caméra. On définit la distance entre deux vues comme la valeur médiane des distances d'un point de contour d'une vue à son correspondant dans l'autre vue.

La figure 2.7 illustre les cartes de contours stables obtenues pour deux vues différentes de la même caméra. La carte de distances de la figure 2.6 est obtenue sur la vue de gauche



(a) (b)

FIG. 2.7 – Contours stables sur deux vues différentes de la même caméra

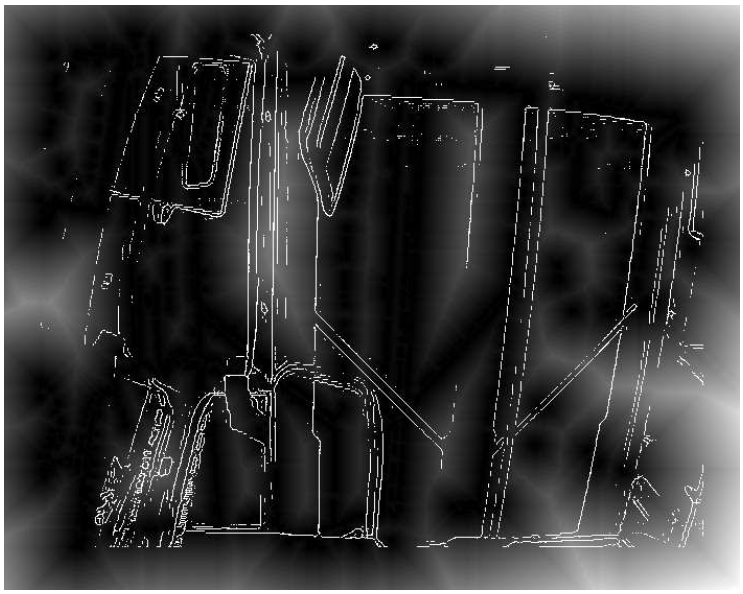


FIG. 2.8 – Superposition de la carte de distances et des contours stables d'une autre vue

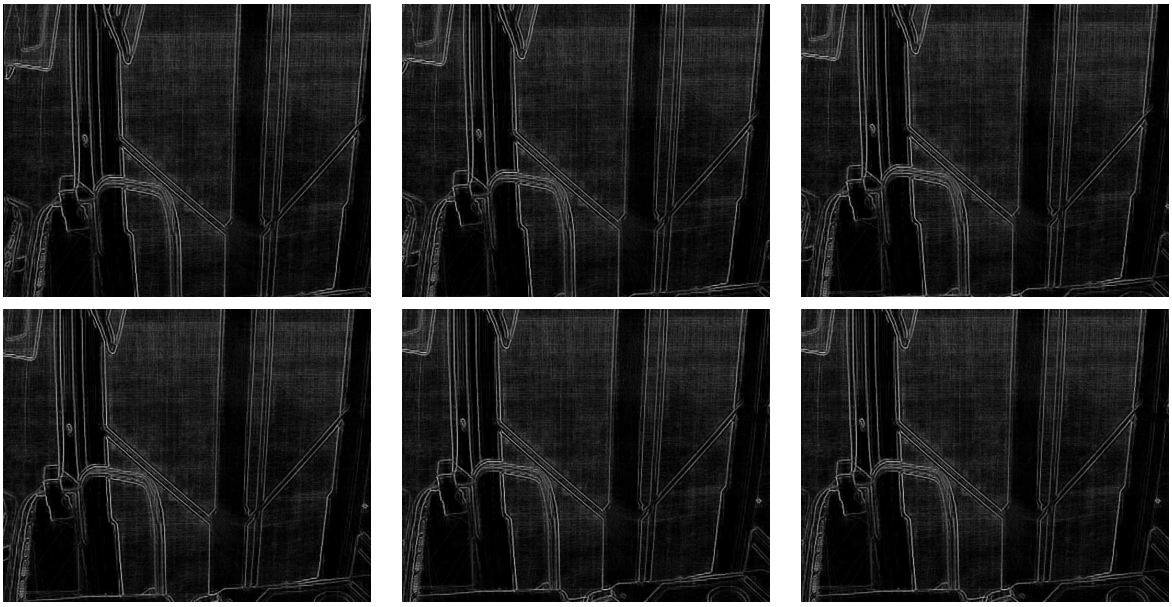


FIG. 2.9 – Exemples de cartes de contours stables générées artificiellement

(a). On peut obtenir la distance d'un point de contour stable de la vue de droite (b) au point de contour stable le plus proche de la vue (a) en récupérant la distance calculée à la position du point de contour stable de (b). Sur la figure 2.6, les contours stables (b) ont été seuillés et superposés à la carte de distances de (a), afin de visualiser le calcul de distance à réaliser. On remarque que beaucoup de contours stables correspondent à des valeurs fortes de la carte de distance. On approxime la distance entre les deux vues de la caméra en calculant la médiane des valeurs de la carte de distance qui correspondent à un point de contour stable de (b).

Bien entendu, il ne s'agit que d'une approximation. En particulier la distance calculée en un point de contour stable de (b) ne correspond pas toujours à la distance réelle entre ce point de contour et le même point de contours du même objet dans la vue (a). Ce phénomène intervient surtout lorsque les deux vues comparées sont à une distance importante l'une de l'autre. Lorsque les deux vues comparées sont très similaires, l'algorithme parvient à déterminer correctement que la distance médiane est très faible, car les valeurs sommées dans la carte de distance correspondent bien aux distances d'un point contour à son point de contour équivalent dans l'autre vue.

Performance de la méthode d'estimation de la distance entre deux vues Afin de quantifier la performance de cette méthode d'estimation de la distance entre deux vues, nous mesurons la distance entre des cartes de contours stables qui ont subi un déplacement en translation généré de manière artificielle.

Les cartes de contours utilisées lors de cette simulation sont en réalité basées sur une carte de contours stables calculée sur une caméra de transport en commun, afin de se rapprocher au plus des conditions réelles. A partir de cette carte de contours stables, 50

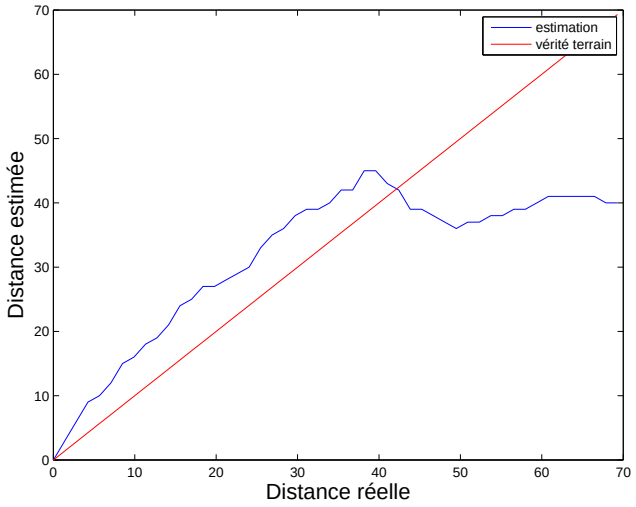


FIG. 2.10 – Distance estimée en fonction de la distance réelle obtenue sur les cartes de contours stables générées

autres cartes sont générées, avec un déplacement en translation allant de 0 à 70 pixels par rapport à la première. La figure 2.9 montre quelques exemples des cartes générées. On notera qu'un agrandissement de la carte originale a été effectué afin de pouvoir réaliser des translations sans avoir à traiter les bords.

La figure 2.10 présente les résultats obtenus lors de cette expérience. La méthode présentée utilisant la carte de distance parvient à estimer la distance entre les deux vues avec une erreur raisonnable jusqu'à environ 40 pixels. Au delà, l'estimation est erronée, mais la distance estimée reste suffisamment élevée pour indiquer que les deux vues sont probablement différentes.

2.1.2.3 Correction des distortions radiales de l'objectif

Les caméras des véhicules de transport en commun doivent couvrir une zone relativement large, pour une meilleure exploitation des images. Comme l'espace dans le véhicule est réduit, les objectifs utilisés ont une courte distance focale, afin de produire un angle de vue élevé. Ce type d'objectif provoque inévitablement des distortions géométriques, qu'il est important de prendre en compte dans les traitements d'images effectués. En effet, à part ces distortions radiales, les différentes orientations possibles de la caméra entraînent à l'image un mouvement apparent que l'on peut raisonnablement supposer linéaire, composé de translations et de rotations. D'autre part, la distortion géométrique due à l'objectif est indépendante de la position de la caméra, et constante. Il est donc logique de corriger cette distortion par un prétraitement, bien que certains auteurs aient proposé d'inclure directement les paramètres de la distortion dans les paramètres à estimer pour le recalage d'images [Fit01].



FIG. 2.11 – Correction de la distortion radiale. à gauche : image originale. à droite : image corrigée

La distortion à corriger est une distortion en barillet : les surfaces paraissent plus grandes proche du centre de l'image et plus petites loin du centre. Cela produit des effets indésirables sur cartes de contours stables. En particulier, les lignes droites paraissent courbées sur les bords de l'image. Le recalage de deux cartes de contours nécessite que les lignes droites restent droites avant et après le mouvement, afin que les contours puissent être associés correctement entre eux.

La distortion radiale est typiquement modélisée par une fonction de mapping $L(r)$, qui modifie la distance r d'un point de l'image au centre de l'image. Un pixel $\boldsymbol{k}$ de l'image sans distortion est transformé en $\boldsymbol{k}'$ dans l'image avec distortion par :

$$\boldsymbol{k}' = \boldsymbol{c} + L(\|\boldsymbol{k} - \boldsymbol{c}\|).(\boldsymbol{k} - \boldsymbol{c}) \tag{2.13}$$

avec $\boldsymbol{c}$ le centre de l'image. La fonction de mapping L est généralement considérée être un polynôme en r^2 d'ordre n_L relativement faible [PK02] :

$$L(r) = 1 + \sum_{i=1}^{n_L} a_{2i}.r^{2i} \tag{2.14}$$

L'estimation automatique des paramètres de distortion à partir d'une image est un problème qui a déjà été abordé depuis longtemps. Elle est généralement basée sur la connaissance *a priori* de caractéristiques connues de la scène, comme par exemple la présence de lignes droites. Ainsi une méthode classique de correction de la distortion de la lentille consiste à chercher la transformation qui permet de retrouver une projection des lignes droites de la scène comme des lignes droites à l'image [SN00], [DF01]. Ces méthodes fonctionnent correctement dès lors que la scène comporte suffisamment de lignes droites. Pour notre application, nous ne nous risquons pas à une estimation erronée des paramètres de distortion. Les coefficients du polynôme ne sont pas estimés automatiquement, mais déterminés de manière empirique pour chaque type de caméras utilisées. L'ordre n_L utilisé est de 1, ce qui donne des résultats satisfaisants. Un exemple de correction est illustré figure 2.11.

D'un point de vue implémentation, la fonction inverse $\boldsymbol{k}' \longrightarrow \boldsymbol{k}$ peut être précalculée pour chaque pixel. Une correction précise doit normalement associer à chaque pixel $\boldsymbol{k}'$ la valeur issue de l'interpolation (par exemple bilinéaire) des pixels autour de la position $\boldsymbol{k}$. Pour des questions de rapidité d'exécution, on considère d'affecter à $\boldsymbol{k}'$ simplement la valeur du plus proche voisin de $\boldsymbol{k}$. Aucune interpolation n'est effectuée, les artefacts dus à la discrétisation de l'image n'étant pas gênants pour le calcul des contours stables. La correction d'une image consiste alors en une simple récupération des coordonnées corrigées dans le tableau précalculé, pour chaque pixel. La correction s'effectue donc très rapidement.

2.2 Mesure de la netteté de l'image

Afin que les images provenant de notre système de vidéosurveillance soit exploitables, il est important que chaque caméra soit correctement focalisée. Les caméras dont nous disposons ne sont pas munies d'autofocus. La mise au point s'effectue manuellement par un opérateur lors de l'installation. L'analyse des images peut permettre de caractériser automatiquement la qualité de la mise au point, en vue d'applications liées à l'installation et la maintenance des caméras. Nous proposons ici deux mesures permettant de caractériser la netteté des images.

2.2.1 Mesures efficaces de la taille des contours

En modélisant l'image nette I_0 par un ensemble de zones d'intensité constantes, séparées par des contours nets, la norme du gradient possède des diracs aux points de contours. On parlera de modèle de contours en escalier. Comme la dérivée d'un signal après convolution par un filtre est égale à la convolution du signal dérivé par le même filtre, le gradient de l'image I est égal à la convolution de la dérivée de I_0 par la PSF f_{PSF} :

$$(I_0 * f_{PSF})' = I_0' * f_{PSF} \tag{2.15}$$

De ce fait, la forme de la PSF apparaît au niveau des points de contours dans la norme du gradient de I.

L'estimation des paramètres de la PSF peut alors être réalisée à partir de la norme du gradient. Les deux mesures de netteté présentées dans la suite se basent sur une estimation de la largeur des contours à partir du gradient de l'image. La première méthode estime l'étalement de la PSF par une adéquation d'un modèle Gaussien + bruit sur le profil médian des points de contours. La seconde méthode approxime cette largeur par un calcul rapide du nombre moyen de pixels appartenant à un profil de contour.

2.2.1.1 Modèle de profil de la norme du gradient

Afin de tenir compte du bruit sur la norme du gradient, créé par l'inadéquation des images réelles par rapport à un modèle de contours en escaliers, on modélise le profil d'un

point de contour par une fonction gaussienne de centre μ et variance σ^2, multipliée par un facteur α et à laquelle s'ajoute une constante β :

$$P(x) = \alpha.\exp(-\frac{(x-\mu)^2}{2.\sigma^2}) + \beta \tag{2.16}$$

Comme on suppose que les paramètres de la PSF sont les mêmes pour tous les points de l'image, les paramètres de notre modèle de profil peuvent être estimés globalement sur tous les profils de contours. On calcule pour cela le profil moyen pour tous les contours, à partir de la norme du gradient de l'image.

Le profil $P_{\boldsymbol{k}}$ pour un point de contour $\boldsymbol{k}$ est obtenu par interpolation bilinéaire des normes du gradient, dans la direction du gradient :

$$P_{\boldsymbol{k}}(x) = b(\|G\|, \boldsymbol{k} + x.\boldsymbol{u_k}) \tag{2.17}$$

avec $\|G\|$ la norme du gradient, $\boldsymbol{u_k}$ le vecteur unitaire dans la direction du gradient en $\boldsymbol{k}$:

$$\boldsymbol{u_k} = \frac{1}{\sqrt{g_h(\boldsymbol{k})^2 + g_v(\boldsymbol{k})^2}}.\begin{bmatrix} g_h(\boldsymbol{k}) \\ g_v(\boldsymbol{k}) \end{bmatrix} \tag{2.18}$$

et b la fonction réalisant l'interpolation bilinéaire sur $\|G\|$, pour le vecteur réel en second argument.

Le profil moyen $\bar{P}$ est alors obtenu en moyennant l'ensemble des profils $P_{\boldsymbol{k}}$ pour chaque point de contour $\boldsymbol{k}$ détecté.

L'estimation des 4 paramètres α, β, σ et μ du modèle de profil consiste en l'adéquation de la courbe $P(x)$ avec le profil moyen $\bar{P}(x)$. L'erreur Q à minimiser est l'erreur quadratique :

$$Q = \sum_{x=-l}^{l} (P(x) - \bar{P}(x))^2 \tag{2.19}$$

avec $2.l + 1 = 25$ la taille du profil choisie pour l'analyse.

L'estimation des paramètres ne peut pas être obtenue par une méthode linéaire. Nous utilisons une minimisation itérative des moindres carrés, réalisée grâce à l'algorithme de Levenberg-Marquardt [Lev44]. Avant de réaliser l'adéquation de $P(x)$ et $\bar{P}(x)$, les valeurs de $\bar{P}(x)$ sont normalisées de sorte que la somme vale 1. D'autre part, l'initialisation des paramètres doit être effectuée avec soin afin que l'algorithme converge correctement vers le maximum global. On choisit comme centre initial μ de la gaussienne la position du maximum de $\bar{P}$, tandis que σ est initialisé à 1. L'amplitude α est quant-à-elle initialisée au maximum de $\bar{P}$, et β à 0.

Sur les figures 2.12 et 2.13 sont représentés le profil moyen et la courbe estimée, obtenues pour des images réelles de transport en commun. L'écart-type de la fonction gaussienne estimée est $\sigma = 1.09$ pour l'image nette, et $\sigma = 1.97$ pour l'image floue.

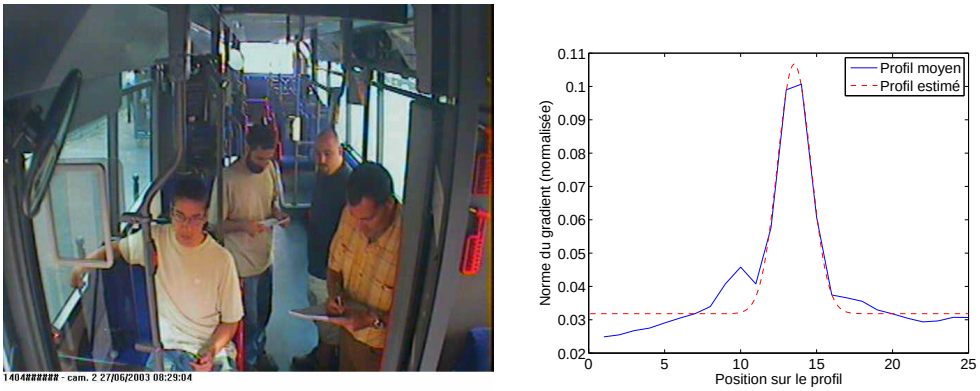


FIG. 2.12 – Adéquation du profil moyen du gradient avec le modèle de contour. Image nette.

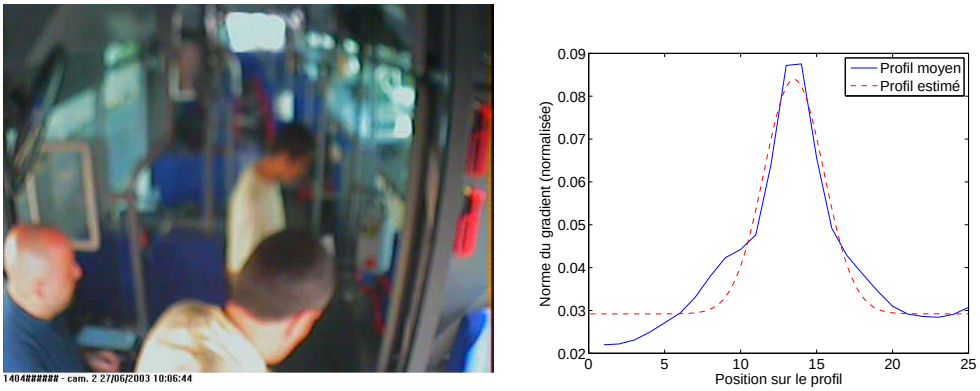


FIG. 2.13 – Adéquation du profil moyen du gradient avec le modèle de contour. Image floue.

2.2.1.2 Approximation de la largeur moyenne des contours

La variance de la PSF gaussienne est fortement liée à la largeur des contours. Cette largeur peut être mesurée sans passer par une estimation de la PSF.

Contrairement à la mesure de flou proposée par [MDWE02], qui détermine aussi la largeur des contours sans estimer les paramètres de la PSF, la mesure proposée ici ne nécessite pas le parcours des profils du gradient, et ne suppose pas l'existence de contours verticaux. En effet, [MDWE02] parcours les profils des contours d'orientation verticale ou quasi-verticale pour ne pas avoir à extraire le profil des contours de directions quelconques.

Une manière de définir la largeur d'un contour est de considérer qu'il s'agit du nombre de points sur son profil dont la norme du gradient est supérieure à un certain seuil. Ce seuil T correspond à la valeur limite de la norme du gradient au delà de laquelle on considère qu'un point appartient à un contour. Il est obtenu automatiquement pour chaque image lors de la détection des points de contours, en supposant une répartition gaussienne centrée des normes du gradient qui ne sont correspondent pas à un contour. Le calcul de la variance des normes à été présenté précédemment, dans les équations 2.3 et 2.4.

Comme on cherche une mesure de netteté qui est globale sur l'image, il n'est pas nécessaire d'extraire le profil de la norme du gradient en chaque point de contour. La mesure proposée est simplement le quotient du nombre N_T de points de normes supérieures à T, par le nombre N_C de points de contours (détectés comme maxima locaux de la norme du gradient supérieurs à T) :

$$L_C = \frac{N_T}{N_C} \tag{2.20}$$

Pour justifier la pertinence de cette approximation, les deux mesures de netteté sont comparées sur une séquence vidéo ou la mise au point varie rapidement, de façon à obtenir différents niveaux de flous. La figure 2.14 illustre la mesure approximée L_C de la largeur moyenne des contours en fonction de la mesure de déviation standard σ du profil moyen. Le coefficient de corrélation entre les deux mesures est de 0.83, ce qui montre qu'elles sont fortement corrélées, et ont donc des performances sensiblement égales pour caractériser la netteté des images. La figure nous apprend aussi que la mesure σ parvient à caractériser une plage plus importante de niveaux de netteté. En effet lorsque $\sigma > 1.3$, L_C n'évolue pas au delà de 3.3. Cette partie de la figure aurait aussi pu être interprétée comme une mesure σ qui soit au contraire très imprécise lorsque $L_C > 3.2$. Les images à partir desquelles nous avons réalisé cette expérience confirment qu'il s'agit bien d'une saturation de l'approximation L_C lorsque l'image est très floue, plutôt que d'un manque de précision de σ. La figure 3.5 présentée dans la suite pour une application d'aide à la mise au point, confrontant les deux mesures, illustre d'ailleurs la saturation de L_C lorsque la mise au point est très mauvaise.

2.2.1.3 Invariance au contenu des mesures de netteté

Afin de valider les mesures de déviation standard σ du contour moyen et de largeur moyenne L_C des contours, une étude de leur invariance par rapport au contenu de l'image est réalisée. Pour cela, on considère les images d'une caméra de surveillance embarquée dans

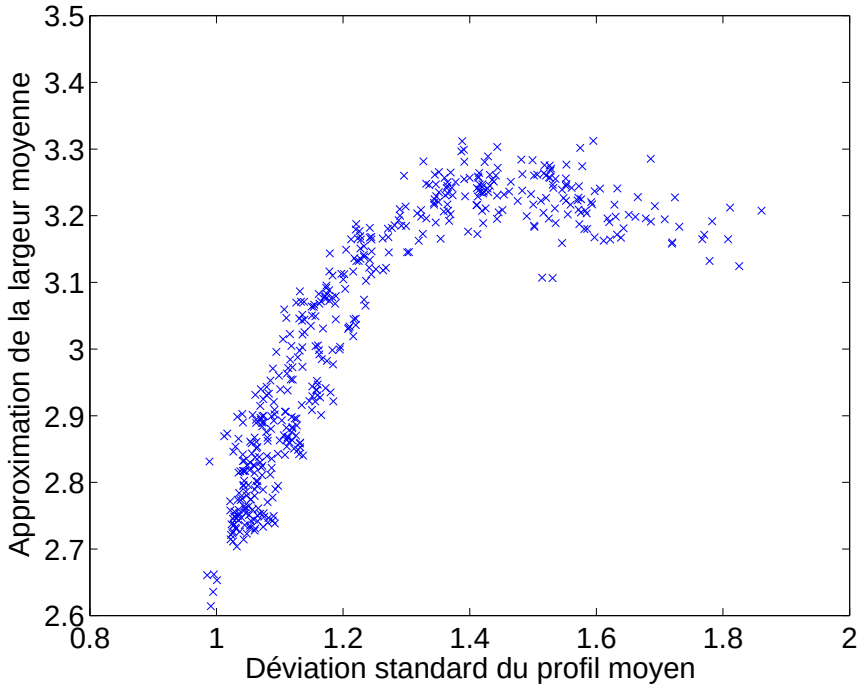


FIG. 2.14 – Corrélation entre les deux mesures de netteté proposées

un véhicule à des instants différents. La mise au point de la caméra ne change pas, mais le contenu de l'image varie fortement, à cause du mouvement des passagers, des changements de luminosités, du paysage extérieur, etc. Les mesures σ et L_C sont alors calculées sur ces images.

Les figures 2.15, 2.16 et 2.17 présentent les mesures obtenues pour trois réglages différents de mise au point. Pour une mise au point correcte ou légèrement déréglée (figures 2.15 et 2.16), les deux mesures de netteté basées sur l'analyse des contours sont invariantes au contenu de la scène, même lorsque la luminosité est radicalement différente. Par contre lorsque la caméra est fortement défocalisée (figure 2.17), les deux mesures varient plus fortement. Quand la mise au point est très mauvaise, et que la scène comporte des zones très éclairées proche de zones sombres, des contours forts apparaissent entre ces zones car le système d'acquisition sature les intensités trop élevées. Les mesures de netteté sont influencées par ce phénomène.

2.3 Obstruction du champ de vision par un objet

La dernière caractéristique qui nous intéresse à propos de l'état du système d'acquisition concerne la bonne visibilité de la scène. Afin de s'assurer que les images renvoyées par les caméras sont bien exploitables, on souhaite détecter automatiquement les cas où un objet gênant vient occulter une partie de la scène. Contrairement à la netteté de l'image,



$\sigma = 0.948594$ $L_C = 2.505546$ | $\sigma = 0.943669$ $L_C = 2.422986$ | $\sigma = 0.863890$ $L_C = 2.362961$ | $\sigma = 0.948629$ $L_C = 2.601722$

FIG. 2.15 – Invariance au contenu des mesures de netteté pour une caméra correctement mise au point



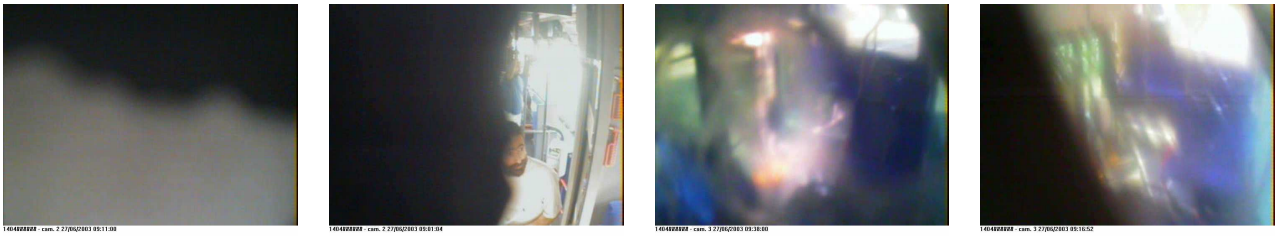
$\sigma = 1.4123$ $L_C = 3.2444$ | $\sigma = 1.2540$ $L_C = 3.1680$ | $\sigma = 1.1653$ $L_C = 3.2050$ | $\sigma = 1.1214$ $L_C = 3.0444$

FIG. 2.16 – Invariance au contenu des mesures de netteté pour une caméra légèrement défocalisée



$\sigma = 1.4807$ $L_C = 3.0669$ | $\sigma = 1.4373$ $L_C = 3.1416$ | $\sigma = 1.3778$ $L_C = 3.1336$ | $\sigma = 1.4054$ $L_C = 3.0228$

FIG. 2.17 – Invariance au contenu des mesures de netteté pour une caméra fortement défocalisée



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

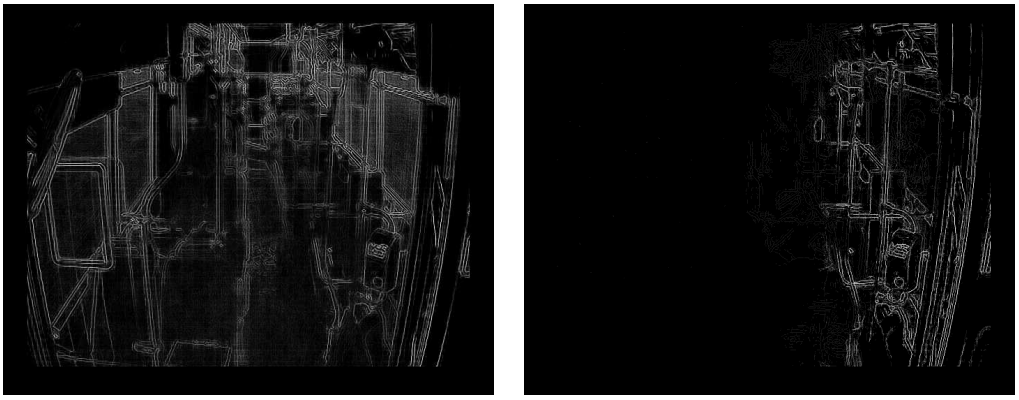
FIG. 2.18 – Différents type d'obstruction de la vue. (a) : opaque totale. (b) : opaque partielle. (c) : semi-transparente. (d) : texturée

l'obstruction de la caméra par un objet n'est pas une caractéristique qui a un sens physique évident. La définition d'une obstruction d'un point de vue image doit être posée correctement, et doit permettre de différencier les objets gênants occultant le champ de vision, des autres objets de la scène tels que les passagers, qui sont justement des objets d'intérêt. Néanmoins, les algorithmes développés pour l'extraction de cette caractéristique doivent rester proches de traitements bas-niveau, afin que la complexité en temps de calcul reste simple. Les objets pouvant gêner le champ de vision peuvent être de toutes sortes, et leur modélisation n'est pas envisageable. Nous nous intéressons donc plus à l'effet de masquage qu'ils produisent plutôt qu'à l'objet lui-même.

2.3.1 Effets possibles d'une obstruction

Des objets de natures différentes provoquent des effets à l'image différents lorsqu'ils sont placés devant la caméra et occultent le champ de vision. Nous ne nous intéressons pas uniquement aux obstructions totales par un objet opaque couvrant entièrement l'objectif, pour lesquelles l'image résultante est noire, et donc facilement identifiable. Les cas d'obstructions que nous souhaitons détecter se différencient selon plusieurs critères :

1. **L'opacité** d'une obstruction peut beaucoup varier. Un objet entièrement opaque (figure 2.18(a)) va évidemment gêner la bonne visibilité de la scène. Mais on souhaite aussi détecter les objets semi-transparents, qui laissent passer une partie de la lumière, mais qui provoque tout de même des images très difficilement exploitables (figure 2.18(c)).
2. **La couverture** est la proportion de l'image qui est affectée par l'obstruction. Il arrive qu'un objet occulte le champ de vision seulement en partie (figure 2.18(b)). L'algorithme de détection doit donc fonctionner même lorsqu'une partie importante de la scène à un comportement normal.
3. **La texture** de l'objet occultant la vue est aussi un critère important. Un objet lisse n'aura pas le même effet à l'image qu'un objet très texturé (figure 2.18(d)). La couleur peut aussi varier.



(a) (b)

FIG. 2.19 – Carte de contours stables d'une vue partiellement occultée

Finalement, les objets peuvent être de tous types et provoquer des effets à l'image qui sont difficilement identifiables, car très variés. Nous cherchons donc plutôt à caractériser l'effet de masquage de la scène qui résulte de la présence d'un objet devant la caméra.

2.3.2 Visibilité des éléments fixes de la scène

On différencie un objet masquant le champ de vision de manière gênante, d'un autre objet de la scène, par deux caractéristiques, qui sont :

1. le temps pendant lequel l'objet reste devant la caméra
2. la proportion de l'image qu'il occupe

Les passagers du véhicule occupent parfois une proportion importante de l'image, mais le temps passé devant l'objectif est supposé relativement court. Il nous faut donc mesurer ces deux caractéristiques, afin de savoir si une partie de la scène est occultée pendant un temps relativement long, par rapport au temps normal de présence d'un objet devant la caméra.

Les contours stables ont déjà été présentés comme une manière de caractériser la position d'une caméra. Ils sont composés des contours des objets fixes de la scène tels que les sièges, les rebords de fenêtre, les barres métalliques, etc. Nous récupérons cette information comme une description de la scène vide, robuste aux objets en mouvement et aux autres variations en illumination. Lorsqu'un objet est présent et statique pendant un temps T devant la caméra, il cache certains points de contours stables. Une carte de contours stables calculée sur une période T sera donc différente lorsqu'un objet obstrue la vue ou non. C'est cette différence que nous souhaitons mesurer, afin de détecter les objets occultant la vue.

2.3.2.1 Proportion visible des contours stables

Sur la figure 2.19(b), la carte de contours stables correspondant à la vue partiellement occultée 2.18(b) est présentée, à titre de comparaison avec la carte de contours stables de

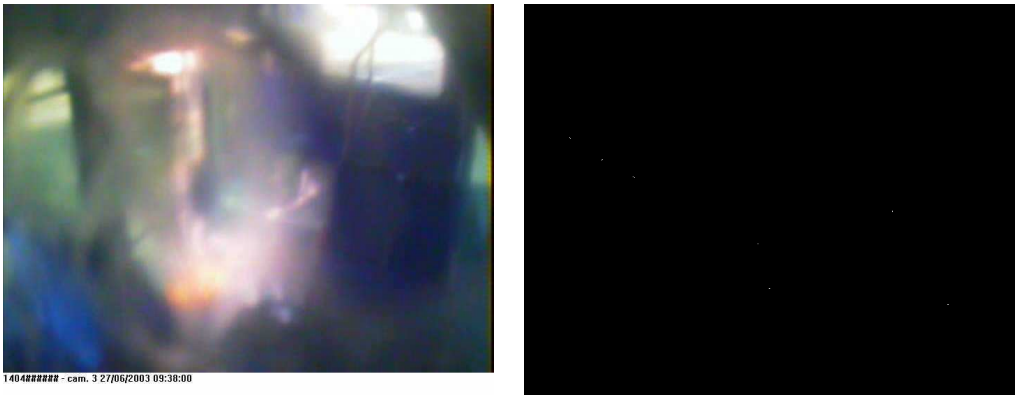


FIG. 2.20 – Carte de contours stables d'une vue occultée par un objet semi-transparent

la même vue sans occultation, figure 2.19(a).

Nous formulons l'hypothèse que les contours stables présents dans la vue de référence, sans occultation, sont en nombre assez important et sont repartis de manière relativement uniforme sur l'image, de façon à ce qu'une occultation même partielle ait une répercution importante sur la proportion de contours stables visibles.

Une mesure possible du *taux d'obstruction* est donc la proportion de contours stables *de référence*, qui sont aussi présents dans la carte des contours stables *courants*. La carte de contours stables de référence est simplement la carte de contours calculée sur une période où la caméra est dans un état idéal, sans problème de mise au point, bien positionnée et sans occultation. La carte des contours stables courants correspond aux contours stables détectés sur la période T précédent l'instant courant. La proportion R_v de contours stables visibles peut être obtenue en seuillant les deux cartes, puis en comptant le nombre N_V de contours stables de référence visible dans la carte courante, parmi les N_R points de contours stables de référence :

$$R_V = \frac{N_v}{N_R} \tag{2.21}$$

Pour les cartes de contours présentées figure 2.19, on obtient une proportion $R_v = 0.23$.

Cas des obstructions semi-transparentes Lorsqu'un objet semi-transparent occulte le champ de vision, il est légitime de se demander comme se comporte la détection des contours stables, sur laquelle est basée notre mesure du taux d'obstruction. Effectivement, l'objet laissant passer une partie de la lumière, la scène reste visible, même si les détails sont fortement dégradés. La figure 2.20 présente la carte des contours stables pour la vue comportant une obstruction semi-transparente totale. Presqu'aucun point de contour stable n'est détecté. Cela s'explique par le fait que les contours de la scène sont très flous, provoquant l'effet d'une forte défocalisation de l'objectif. Le bord des objets ne possède pas une norme de gradient très élevée par rapport aux autres points de l'image, ce qui fait que la détection des contours échoue. De ce fait, la mesure de la proportion de contours



FIG. 2.21 – Découpage en zones de l'image pour la mesure d'obstruction

stables visibles R_V fonctionne très bien pour ce type d'obstruction.

2.3.2.2 Amélioration de la détection des obstructions partielles

La mesure R_V, équation 2.21, dépend trop de la couverture de l'obstruction. Les obstructions partielles faibles provoqueront une proportion de contours stables visibles relativement forte, et pourront ne pas être détectées. Pour améliorer cette mesure, nous proposons de découper l'image en n_Z zones, et de calculer la proportion R_v pour chacune de ces zones. La nouvelle mesure R_V^{min} est alors la proportion minimum parmi les proportions R_v^i de contours stables visibles de chaque zone.

$$R_V^{min} = \min_{i=1...n_Z} R_v^i \tag{2.22}$$

Avec un découpage de l'image en $n_Z = 3 \times 3$ réparties comme sur la figure 2.21, les proportions R_v^i mesurées pour chaque zones i au moment de l'obstruction partielle de la figure 2.18(b) sont présentées dans la matrice suivante :

$$\begin{bmatrix} 0 & 0.10 & 0.64 \\ 0 & 0.20 & 0.91 \\ 0 & 0.30 & 0.94 \end{bmatrix} \tag{2.23}$$

La proportion minimum, obtenue pour l'une des zones de gauche, est $R_v^{min} = 0$, ce qui permet de détecter l'obstruction sans difficulté.

2.4 Conclusions sur les mesures de qualité du système d'acquisition

Dans ce chapitre, des caractéristiques relatives à l'état du système d'acquisition ont été définies, et des algorithmes permettant de les extraire ont été proposés. Il est maintenant

possible au système de surveillance de connaître l'état des caméras directement à partir des images acquises. Les trois propriétés étudiées ont été la position d'une caméra, la qualité de la mise au point, et l'occultation de la scène par un objet gênant. Pour chacune, nous avons développé des algorithmes permettant d'extraire des mesures pertinentes à la propriété étudiée. Ces mesures sont stables dans le temps, malgré les fortes variations dans la scène, relatives aux mouvements et à l'illumination.

Selon les applications que nous présenterons dans le chapitre suivant, le fonctionnement en temps-réel peut être requis. C'est pourquoi les mesures de distance médiane aux contours et de largeur moyenne des contours sont proposées pour caractériser la position de la caméra et la netteté des images respectivement. Elles peuvent être vues comme des approximations des mesures plus précises, et plus pertinentes d'un point de vue physique, des paramètres de recalage entre cartes de contours et des paramètres du profil moyen du gradient des contours.

Le chapitre suivant présente des applications utilisant directement ces mesures de l'état du système, pour les deux familles de besoins que sont l'autosurveillance des caméras et l'aide à l'installation des caméras.

Chapitre 3

Applications liées à l'état du système d'acquisition

Sommaire

3.1 Autosurveillance des caméras 60
3.1.1 Extraction des états de référence 60
3.1.1.1 Apprentissage en continu 61
3.1.2 Détection de changement dans les mesures 62
3.1.2.1 Détection d'une variation forte de la mesure 62
3.1.2.2 Lissage temporel des mesures 65
3.1.2.3 Résultats de la détection de défauts 66
3.2 Aide à l'installation 67
3.2.1 Aide à la mise au point 68
3.2.2 Aide au positionnement 70
3.3 Conclusions sur les applications développées 72

Dans le chapitre précédent, des outils permettant de caractériser l'état du système d'acquisition ont été présentés. Ils permettent d'extraire de la vidéo des informations concernant la position de la caméra, la netteté de l'image et la visibilité de la scène. Deux familles d'applications utilisent directement ces informations, en vue de garantir une bonne exploitation des images de surveillance.

Tout d'abord, nous nous intéresserons aux applications d'**autosurveillance des caméras**, dont le but est de s'assurer continuellement que les caméras ne se dérèglent pas pendant leur fonctionnement, soit à cause d'une malveillance, soit à cause de phénomènes mécaniques telles que les vibrations du véhicule.

Le second type d'applications concerne l'**aide à l'installation des caméras**. L'analyse des images va permettre d'indiquer à un opérateur humain la qualité du réglage de la

caméra en temps réel, afin de le guider vers le réglage optimal.

Les priorités pour le choix des algorithmes sont différentes pour les deux types d'applications. Pour l'autosurveillance de caméras, on privilégiera des algorithmes performants en détection, afin de limiter les fausses alarmes et de détecter correctement les anomalies. Pour l'aide à l'installation des caméras, on préférera au contraire des algorithmes rapides, capable d'effectuer une analyse des données vidéo avec une cadence rapide.

3.1 Autosurveillance des caméras

Le système de vidéosurveillance développé possède une fonction d'autosurveillance, qui lui permet de détecter automatiquement toute anomalie qui pourrait se produire pendant le fonctionnement. Certaines anomalies comme typiquement les défaillances du système d'exploitation ou encore un câble débranché, peuvent être détectés de façon électronique, en vérifiant en permanence la présence d'un signal électrique. Les anomalies que nous traitons ici sont celles qui ne peuvent pas être détectées directement de façon électronique, mais pour lesquelles l'analyse des images peut permettre leur détection de façon automatique. En partant des outils d'analyse de l'état des caméras présentés dans le chapitre précédent, nous décrivons ici le système d'autosurveillance des caméras, pour les anomalies concernant la position de caméra, la netteté des images et la visibilité de la scène. L'autosurveillance est décomposée principalement en deux étapes, qui sont l'apprentissage d'un état de référence, représentant l'état idéal dans lequel devrait se trouver la caméra, puis la détection de changement fort de l'état par rapport à cet état de référence.

3.1.1 Extraction des états de référence

L'extraction des états de référence est un problème à part entière. Il s'agit d'apprendre automatiquement l'état idéal de la caméra, à partir des caractéristiques de position, de netteté, et de visibilité. Comme le système doit fonctionner de la manière la plus autonome possible, il doit apprendre par lui-même quel est son état idéal. Il est difficile de prévoir la période pendant laquelle la caméra est dans son état de référence. On peut bien sûr considérer qu'après l'installation, la caméra est parfaitement réglée et que l'apprentissage peut être réalisé à ce moment. Mais l'apprentissage est une tâche qui doit être effectuée pendant un temps relativement long, afin que l'on puisse considérer le plus de cas possibles pour l'état de référence.

L'apprentissage de l'état de référence pour les trois caractéristiques auxquelles nous nous intéressons se ramène à l'extraction d'une carte des contours stables de référence. En effet, pour la caractéristique de position de caméra, la mesure concernée est directement liée à cette carte, que ce soit pour le calcul de la carte de distance ou pour une estimation de mouvement apparent plus précise. Pour la caractéristique de bonne visibilité de la scène, le taux d'obstruction est aussi calculé en fonction de la carte des contours stables de référence. Seule la caractéristique de netteté de l'image n'est pas directement liée aux

contours stables, mais elle est basée sur la norme du gradient d'images de référence, qui doit être de toute façon calculée pendant l'extraction des contours stables de référence.

L'apprentissage de la carte de contours stables de référence doit être fait à partir d'un grand nombre d'images, afin que les contours stables soient proprement détectés, avec le moins de fausses détections possible. La tâche n'est pas facile, car il arrive régulièrement que suite à l'installation, le véhicule reste à une position donnée, par exemple dans un dépôt de véhicules, pendant un temps très long. Si l'apprentissage des états de référence est réalisé pendant cette période, cela revient à ce qu'il n'y ait finalement qu'une seule image d'exemple pour l'apprentissage, avec une certaine luminosité, un certain paysage extérieur, etc. La carte de contours stables contiendrait alors des contours marqués provenant du paysage extérieur dans les parties vitrées.

3.1.1.1 Apprentissage en continu

Le temps d'apprentissage doit être long, afin d'intégrer le plus grand nombre de cas d'images possible. Dans le même temps, il faut éviter au maximum les risques de déréglages de la caméra pendant la phase d'apprentissage. Pour répondre à ces deux critères, nous réalisons un apprentissage en continu, qui a lieu en permanence pendant le fonctionnement du système de surveillance, mais qui s'arrête tout de même lorsque certaines conditions ont lieu.

Premièrement, lorsqu'une anomalie est détectée sur une caméra, les images correspondantes ne sont bien entendue pas utilisée pour la mise à jour. Deuxièmement, les images trop sombres ne sont pas non plus prises en compte, car la détection des contours y est moins efficace. Enfin troisièmement, lorsque le bus est immobile, la mise à jour de la référence est aussi stoppée, afin que des contours du paysage extérieur n'apparaissent pas sur la carte des contours stables.

Nous avons décrit précédemment la méthode d'obtention de la carte des contours stables courants, équation 2.5, qui opère une moyenne glissante en simulant une moyenne pondérée des cartes de contours en fonction de leur date d'apparition. L'obtention de la carte des contours stables de référence est réalisée de la même manière, mais avec un temps d'intégration T beaucoup plus long. Ce temps d'intégration doit être suffisamment long pour obtenir une détection propre, mais doit aussi permettre la prise en compte de changements persistants de la scène vide, tels que l'apparition d'un objet statique (une publicité par exemple) ou sa disparition. Avec un temps T équivalent à une journée entière, ce type de changements persistants est pris en compte correctement, tout en laissant la référence suffisamment stable pour permettre une détection efficace des anomalies, grâce au temps d'intégration beaucoup plus court des contours stables courants.

Pour la mesure de netteté de référence L_C^{ref}, on réalise aussi une intégration temporelle des mesures L_C, équation 2.20, pour chaque image, afin de réduire l'effet du bruit et de toute mesure qui pourrait être erronée. L'intégration est aussi réalisée par une moyenne pondérée sur une période T :

$$L_C^{ref}(t) = \frac{1}{T}\Big((T-1).L_C^{ref}(t-1) + L_C(t)\Big) \tag{3.1}$$

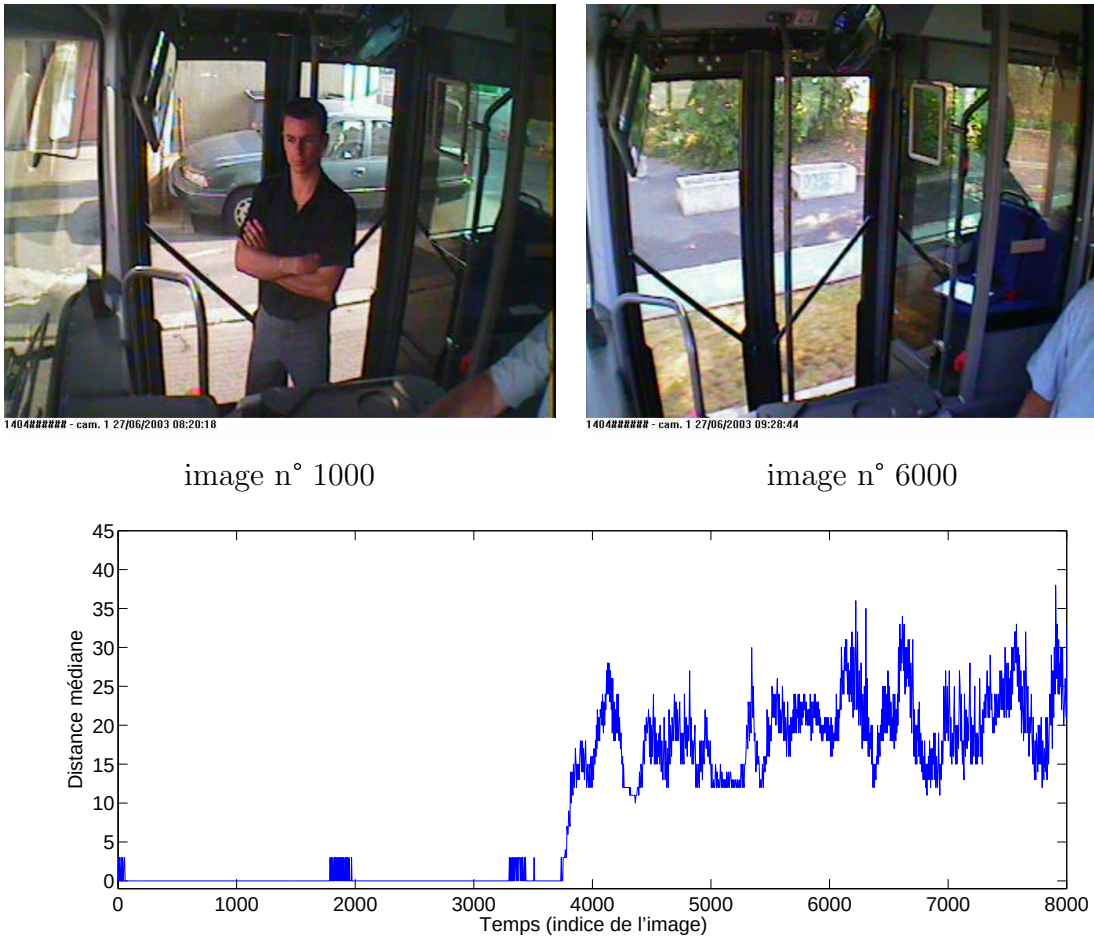


FIG. 3.1 – Évolution de la mesure de décadrage (distance médiane entre contours stables de référence et courants calculée par la carte de distances aux contours stables courants) lors de l'apparition d'un décadrage de la caméra à l'image 3754

3.1.2 Détection de changement dans les mesures

La détection des anomalies de caméras pendant le fonctionnement du système de surveillance peut être assimilée à un problème de détection de changement fort dans une mesure.

3.1.2.1 Détection d'une variation forte de la mesure

L'importance de ce changement peut être déterminée de façon très pragmatique, lorsque la mesure de la caractéristique correspond à une quantité physique facilement compréhensible. L'algorithme de détection d'un déplacement de la caméra en est le meilleur exemple. La taille du décadrage de la caméra par rapport à sa position de référence est la quantité qui est directement mesurée, et la limite au delà de laquelle ce décadrage est considéré comme une anomalie peut être directement spécifiée. On choisira par exemple de considérer les



image n° 500 image n° 1400

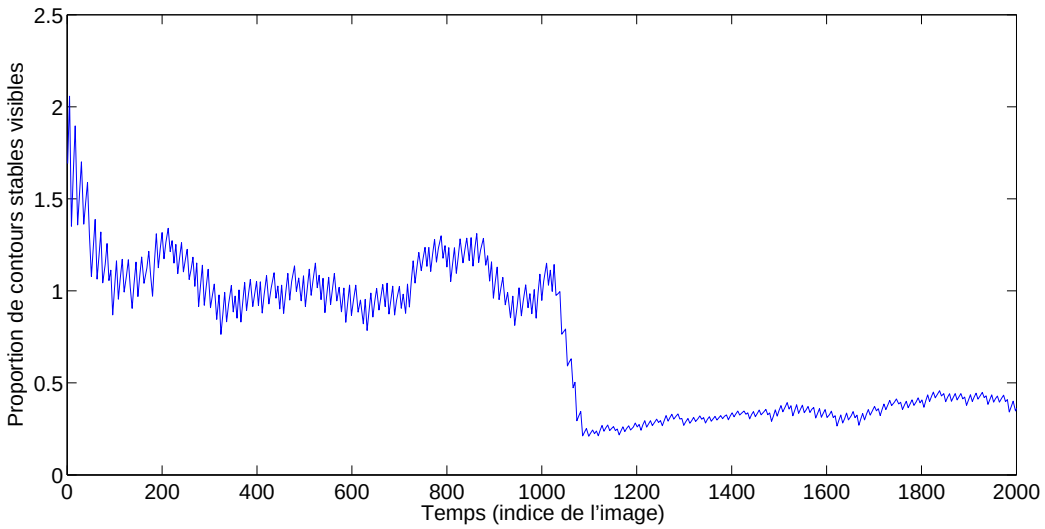


FIG. 3.2 – Évolution de la mesure d'obstruction du champ de vision (proportion de contours stables de référence visibles) lors de l'apparition d'un défaut

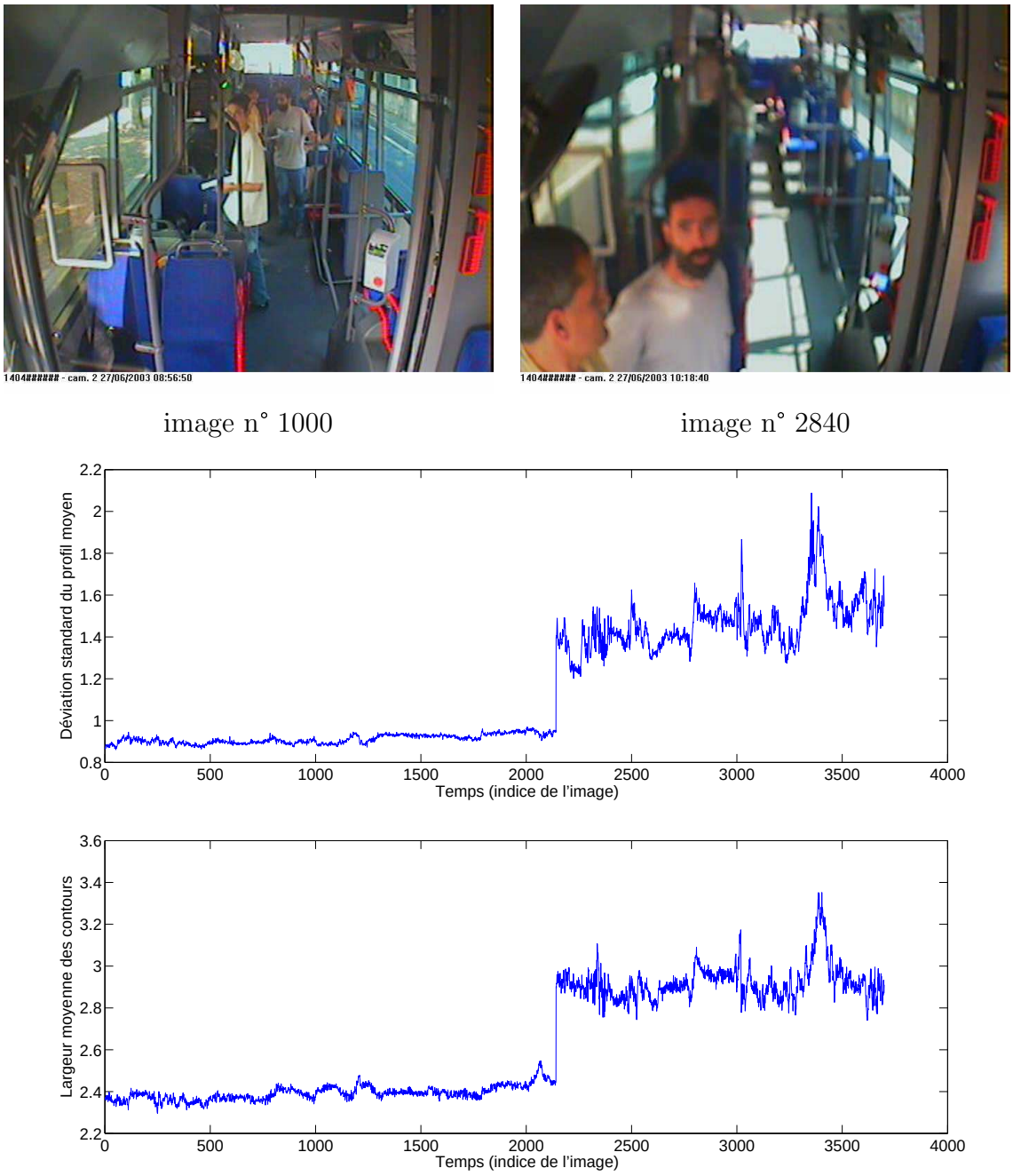


FIG. 3.3 – Évolution des mesures de netteté lors de l'apparition d'un défaut de mise au point

translations du champs de vision au delà de 5% de la taille de l'image comme des anomalies. La figure 3.1 montre l'évolution de la mesure de distance médiane aux contours stables lors de l'apparition d'un décadrage. On observe que la mesure est très stable en l'absence de défaut malgré les perturbations dans la scène, et réagit correctement à l'apparition de l'anomalie.

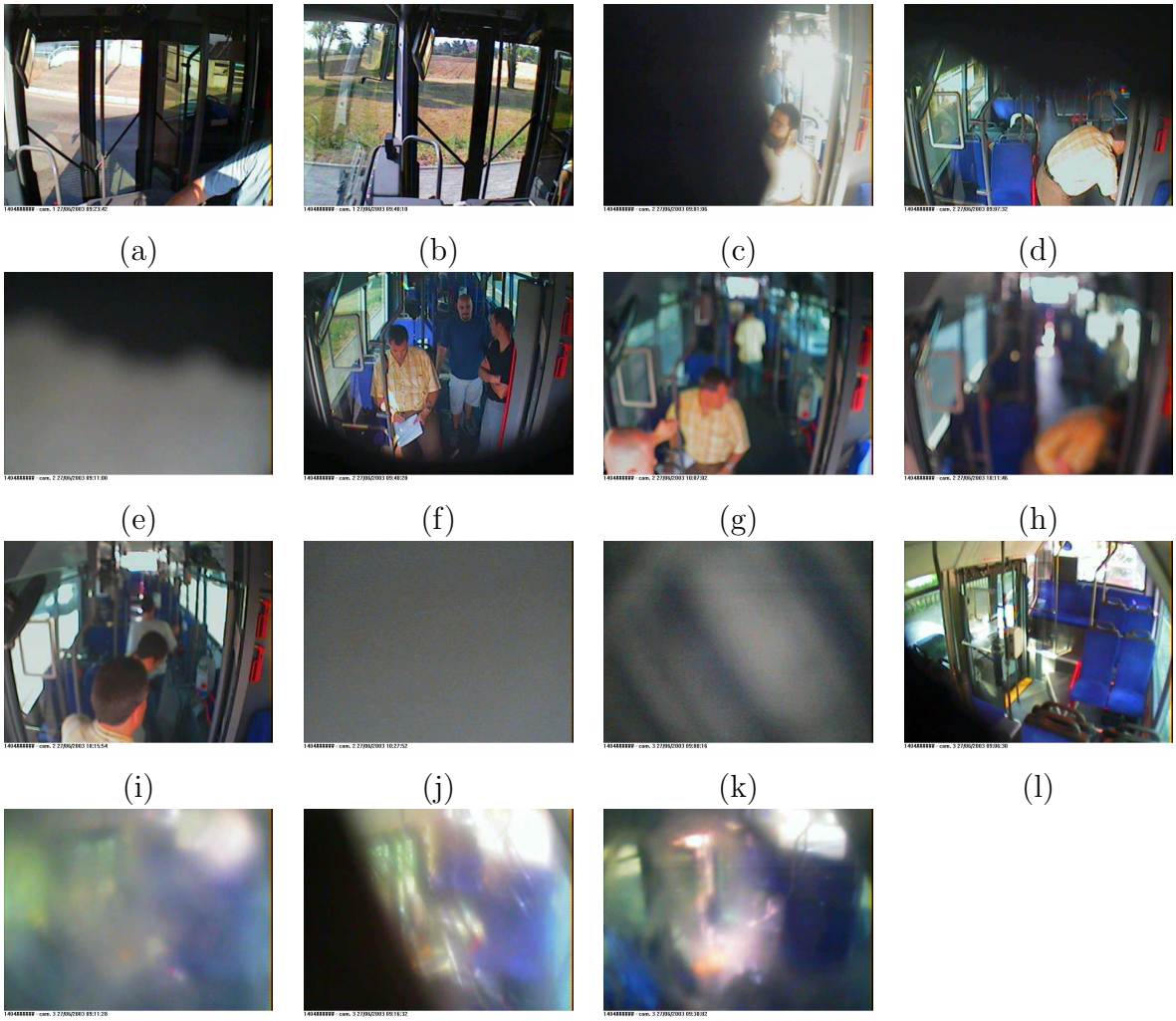
La détection d'une obstruction de la caméra est plus délicate, car la mesure du taux d'obstruction est plus bruitée que celle de la taille du décadrage. En effet, les points de contours stables courants détectés ne sont pas toujours exactement identiques aux points de contours stables de référence, même en l'absence d'anomalies. Nous cherchons donc à modéliser les variations normales de la mesures, en l'absence d'anomalies, en considérant la variance du taux d'obstruction au cours du temps. On considère alors qu'une anomalie intervient lorsque la mesure R_V^{min}, équation 2.22, baisse de plus de trois fois son écart-type. La figure 3.2 illustre un exemple d'évolution de la mesure lors de l'apparition d'un défaut d'obstruction. On remarquera sur cette figure que la mesure peut être supérieure à 1 bien qu'il s'agisse en théorie d'une proportion. Cela provient de la manière de calculer cette mesure, qui est en pratique une approximation de la proportion de contours stables visibles. Le nombre de contours stables courants peut en effet être supérieur au nombre de contours stables de référence. Une intersection entre les deux cartes de contours stables pourrait résoudre ce problème, mais introduirait une trop forte sensibilité de la mesure aux petits décadrages de la caméra qui peuvent se produire à cause des vibrations du véhicule.

Une méthode similaire est utilisée pour la détection d'un déréglage de la mise au point. La figure 3.3 montre l'évolution des mesures de netteté (déviation standard du modèle de contour et approximation de la largeur moyenne des contours) au moment de l'apparition d'un défaut de mise au point.

3.1.2.2 Lissage temporel des mesures

Le but de la surveillance des caméras n'est pas de localiser une anomalie précisément dans le temps. Il est préférable de rapporter un défaut au système seulement lorsqu'on est sûr qu'un problème est intervenu sur l'une des caméras, plutôt que de risquer de rapporter des fausses alarmes. Lorsqu'un défaut est rapporté au système, un opérateur est prévenu et pourra vérifier l'état des caméras, ou la vidéo qui aura été enregistrée. Comme le temps réel n'est donc pas requis pour la détection des anomalies, un lissage temporel de la mesure instantanée du netteté est réalisé, afin de supprimer au plus l'effet des mesures erronées. On réalise pour cela une moyenne glissante de la mesure de netteté, d'une manière similaire à la moyenne glissante des cartes de contours présentée équation 2.5. La période T contrôle le temps de mise à jour de la mesure, et dépend donc directement du temps que l'on souhaite tolérer pour la détection d'un défaut. Plus ce temps est long, plus le risque de fausses détections intempestives de défauts est faible. Nous choisissons une période T équivalente à trois minutes afin d'être suffisamment robuste à des images pour lesquelles la mesure de netteté pourrait être erronée.

Les mesures utilisées pour l'obstruction et la position de la caméra sont déjà lissées temporellement, car basées sur les cartes de contours stables.



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

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

(i) (j) (k) (l)

(m) (n) (o)

FIG. 3.4 – Défauts présents dans les séquences de test

3.1.2.3 Résultats de la détection de défauts

Les algorithmes de détection de défauts ont été validés sur des séquences vidéos réelles issues de caméras embarquées dans un véhicule de transport en commun. Une campagne d'acquisition a été spécialement réalisée afin de disposer de 9 heures de séquences vidéos, réparties en 3 heures pour chacune des 3 caméras du véhicule. Les défauts que nous souhaitons détecter ont été simulés pendant ces 9 heures. Ces 9 heures sont composées de 7 heures de vidéo sans défaut, et de 2 heures de vidéo avec défauts. La figure 3.4 illustre les différents défauts qui ont été simulés. Les séquences de test comprennent plus précisément les défauts suivants :

- 2 décadrages légers de caméra (a)(b)
- 3 obstructions partielles et opaques du champ de vision (c)(d)(l)

- 1 décadrage léger combiné à une obstruction partielle opaque. (f)
- 3 obstructions totales du champ de vision (e)(j)(k)
- 3 mauvaises mises au point, avec des niveaux de netteté différents (g)(h)(i)
- 2 obstructions semi-transparentes totales (m)(o)
- 1 obstruction totale composée d'une partie opaque et d'une partie semi-transparente (n)

Afin de s'approcher au mieux des situations que l'on peut rencontrer dans un contexte réel de fonctionnement, nous avons simulé plusieurs situations qui pouvaient mettre en difficulté l'analyse des images. En particulier, le nombre de personnes devant la caméra varie régulièrement, allant d'une scène vide jusqu'à la simulation d'un véhicule bondé en heure de pointe. En terme d'illumination, nous disposons de séquences prises lors du passage sous un tunnel, changeant radicalement le comportement de la caméra pendant environ deux minutes. La campagne d'acquisition a été réalisée entre 8h et 11h du matin, en été, ce qui fait que nous ne disposions pas de séquences de nuit. Néanmoins, d'autres séquences vidéos, réelles cette fois, acquises ultérieurement à partir d'autres véhicules comportent quant-à-elle des séquences vidéo de nuit. Celles-ci ne comportent par contre pas de défaut de caméra à détecter. Ces différentes conditions permettent de tester la robustesse de la détection de défauts.

Les résultats obtenus sont très bons, vu que la totalité des défauts de caméras dans les séquences disponibles ont été détectés. Dans les 9 heures de séquence analysées, correspondant à plus de 150000 images, aucune fausse alarme n'est apparue.

Il est toutefois à noter que ces séquences vidéo ont été acquises le matin, et qu'elle n'incluent donc pas de séquences de nuit. Le système développé n'est pourtant pas exempt d'imperfection. Il arrive effectivement qu'un défaut soit détecté comme un autre défaut. En particulier, les obstructions semi-transparentes sont visuellement très similaires aux mauvaises mises au point, et les deux types de défauts sont donc très corrélés. Néanmoins, en utilisation courante, le type de défaut détecté n'est pas l'information la plus importante. Du point de vue de l'utilisateur, il est beaucoup plus important d'obtenir de bonnes performance en détection, avec un faible taux de fausse alarme.

Les séquences dont la luminosité est très faible ne posent pas de problème de fausse alarme, dès lors que l'image n'est pas complètement noire, auquel cas les contours ne peuvent pas être détectés correctement. Cela nous conforte sur la qualité des mesures, qui ont été conçues de façon à prendre en compte différents types d'illumination. En particulier, la détection des points de contour avec seuillage automatique fonctionne bien pour des faibles luminosités.

3.2 Aide à l'installation

L'autre catégorie d'applications liée à l'état du système d'acquisition concerne l'aide à l'installation des caméras. Lorsqu'il s'agit d'installer le système de surveillance pour un réseau de véhicules, un *expert sécurité* a pour mission de décider du nombre et de la

position des caméras afin que le système surveille efficacement le véhicule. L'expert veille aussi à la bonne qualité des images délivrées, en terme de couleurs, de luminosité et de netteté. En pratique, les véhicules à installer sont tous similaires, et l'expert n'assure la bonne installation du système que sur un seul véhicule. D'autres techniciens s'occupent de l'installation des autres véhicules du parc, en reproduisant au mieux ce que l'expert a décidé. Cela crée parfois des problèmes, lorsque l'installation n'est pas correctement vérifiée. Nous proposons ici d'utiliser l'analyse d'image comme outil pour aider à l'installation des caméras, pour deux caractéristiques primordiales de la caméra, qui sont la qualité de la mise au point et la qualité du positionnement. Pour la qualité du positionnement des caméras, on utilisera des informations extraites du premier véhicule, réglé par un expert, afin de comparer la vue de la caméra d'un véhicule avec la vue de ce premier véhicule.

3.2.1 Aide à la mise au point

L'aide à la mise au point est une application dont le but est de permettre un réglage optimal de la distance lentille-capteur des caméras, sans avoir besoin de voir l'image acquise au moment du réglage. Le principal avantage d'une telle application est qu'elle supprime le besoin d'installer un écran de surveillance pendant l'installation, et il facilite donc l'installation du système.

Le principe de l'aide à la mise au point est d'émettre un signal sonore dont les paramètres varient en temps réel en fonction de la mesure de netteté. L'opérateur règle la mise au point en fonction du signal sonore, de manière asservie, jusqu'à atteindre le réglage optimal.

Un signal sonore sinusoïdal $J(t)$, dont la fréquence varie de façon inversement proportionnelle à mesure de netteté L_C (équation 2.20), est généré :

$$J(t) = A.sin(2\pi.f(L_C).t) \tag{3.2}$$

avec $f(L_C)$ la fréquence du signal, réglée de manière à produire un son audible, dans une gamme suffisamment large pour que l'oreille puisse discriminer facilement les différents réglages de mise au point. Pour notre application, la fréquence suivante est adéquate :

$$f(L_C) = (200 + 1/L_C * 1000) \tag{3.3}$$

L'évolution des mesures de netteté pendant une expérience de réglage de mise au point sur la caméra couloir d'un véhicule est présentée sur la figure 3.5. L'expérience a consisté à tourner rapidement la bague de mise au point de la caméra dans les deux sens pendant quelques secondes, autour de la position optimale, afin d'obtenir différents niveaux de netteté dans la séquence vidéo. La bague a ensuite été réglée à la main sur une position qui semblait procurer l'image la plus nette à l'image. Comme on peut le constater sur les courbes de mesures, le réglage final, obtenu par rapport à l'image renvoyée en direct sur un écran, ne correspond pas au réglage le plus net possible. L'application d'aide à la mise au point aurait permit d'obtenir un réglage plus fin, grâce au signal audio.

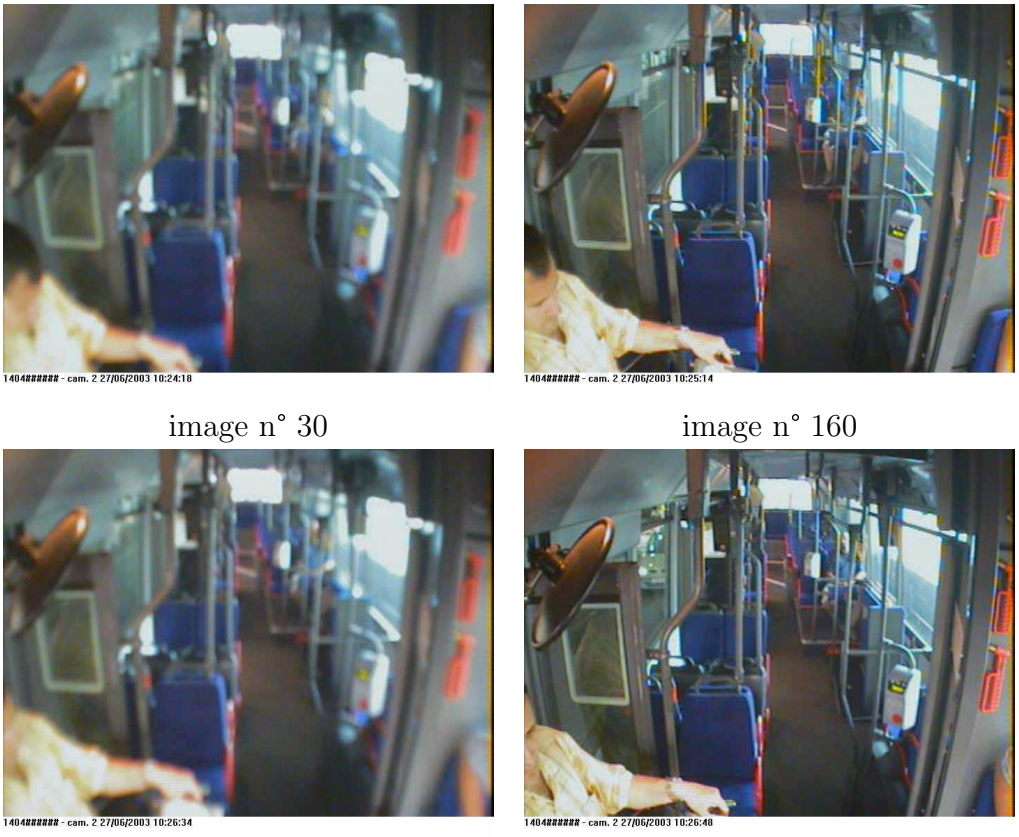


image n° 30 image n° 160

image n° 345 image n° 380

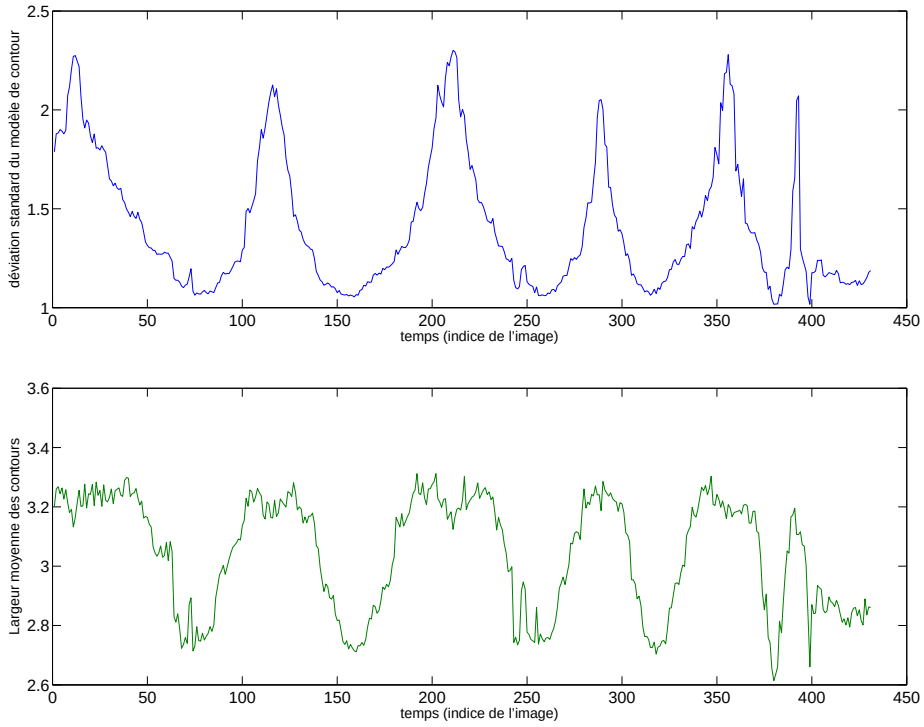


FIG. 3.5 – Évolution des mesures de netteté pendant un réglage de mise au point

On observe que les deux mesures sont fortement corrélées, ce qui nous rassure sur l'utilisation de la largeur des contours L_C comme mesure de netteté, malgré sa plus faible précision dans les réglages très flous. Le comportement de la mesure L_C est par ailleurs très bon aux alentours du réglage optimal, ce qui la rend très utilisable pour notre application d'aide à la mise au point.

3.2.2 Aide au positionnement

L'aide au positionnement est une application basée sur le même principe que l'aide à la mise au point. L'opérateur est aidé pendant l'installation de la caméra par un signal sonore, dont les paramètres varient en fonction de l'orientation de la caméra. Il cherche la position optimale de la caméra, qui est indiqué par le maximum de la fréquence d'un signal sonore. Cette fréquence varie en fonction de la mesure de distance médiane (*cf.* section 2.1.2.2) entre une carte de contours apprise sur une caméra réglée correctement dans un véhicule similaire, et des contours courants du véhicule en train d'être installé. La mesure porte sur les contours de chaque image, et non sur les contours stables, afin que l'intégration temporelle n'entraîne pas de décalage entre la fréquence du son et les mouvements réalisés par l'opérateur. D'autre part, les contours du premier véhicule utilisé pour le calcul de la distance médiane proviennent d'une seule image, prise lorsque la caméra est bien réglée. Ils contiennent donc *a priori* des contours du paysage extérieur. Il n'est pas raisonnablement envisageable de calculer la distance médiane en utilisant une carte de contours stables issue du premier véhicule, car le calcul de celle-ci requiert que le véhicule soit en mouvement, et demande un temps d'intégration suffisamment long. Ce n'est pas envisageable dans les cas où l'installation du parc de véhicule doit être réalisée rapidement.

La mesure de distance va donc porter sur deux cartes de contours assez différentes. On suppose que les points de contour du paysage sont en nombre assez restreint par rapport aux contours du véhicule pour que la distance médiane soit suffisamment robuste.

La figure 3.6 présente un exemple de simulation d'aide au positionnement. Les contours d'une vue de la caméra couloir (figure 3.7(a)) ont été extraits, et utilisés comme position de référence pour le réglage d'une autre caméra. On utilise en réalité pour cette simulation la même caméra, dans le même véhicule, mais à un instant différent afin que les conditions de la scène changent légèrement. La séquence de simulation présente donc la scène qui subit un mouvement apparent, lors d'un réglage de la caméra. La mesure de distance médiane est illustrée ici. Elle atteint son minimum lorsque les deux vues (vue de référence et vue courante) coïncident au mieux. La distance de 30 pixels obtenue à l'image 12 comme minimum est relativement élevée pour deux raisons. Tout d'abord, la faible cadence d'images de la séquence de simulation est telle que l'image correspondant à la position optimale de la caméra n'a pas été acquise. Par ailleurs, la différence de contenu entre les deux images de la figure 3.7 est importante, principalement à cause des personnes présentes dans le véhicule et d'une porte intérieure ouverte dans la vue (b). On observe toutefois sur la figure 3.7 que la position optimale estimée est très proche de la meilleure position atteignable. La proxi-



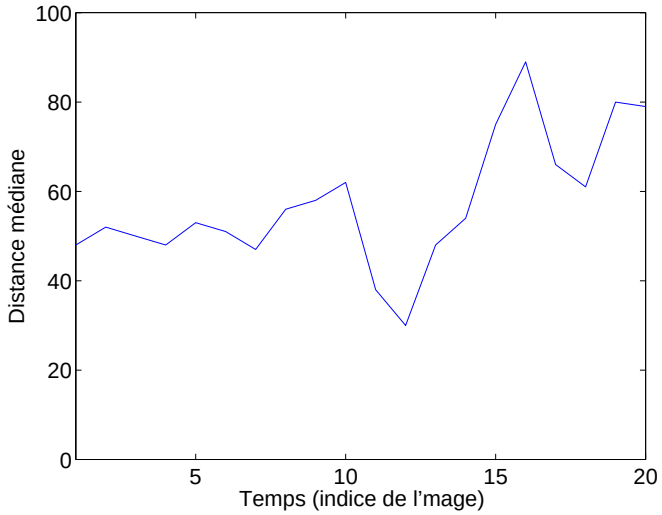
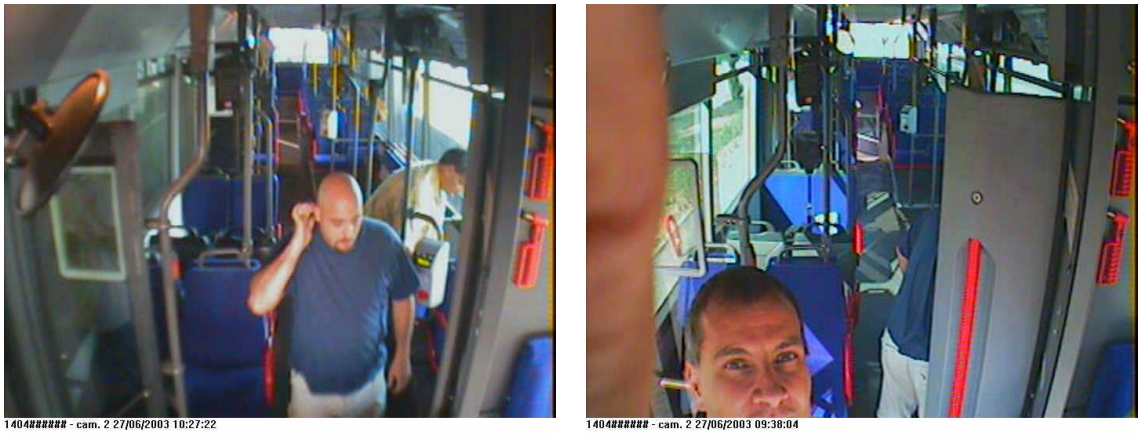


FIG. 3.6 – Évolution de la distance médiane d (en pixels) entre contours pendant un réglage de position, par rapport à la position de référence figure 3.7



(a) (b)

FIG. 3.7 – (a) : Position de référence pour la simulation d'aide au positionnement. (b) : Image 12, position optimale de la caméra par rapport à la mesure de distance médiane (distance de 30 pixels)

mité entre les deux vues peut être observée aux alentours de la vitre arrière du véhicule, positionnée en haut au centre dans l'image.

3.3 Conclusions sur les applications développées

Les applications développées à partir des mesures caractérisant le système d'acquisition viennent d'être présentées. Elles répondent en partie à deux besoins de l'industriel, qui sont d'apporter à l'exploitant un allègement de la maintenance du système grâce aux fonctions d'autosurveillance des caméras, et une facilité d'installation des caméras grâce à un guidage sonore. Les mesures caractérisant l'état du système d'acquisition que nous avons développées et présentées au cours du chapitre précédent prouvent ici leur utilité et leur performance pour des applications concrètes.

Ces fonctionnalités donnent une réelle valeur ajoutée à l'ensemble du système de surveillance, ce qui nous conforte dans le potentiel d'améliorations que peut apporter l'analyse d'images au milieu industriel. Nous avons effectivement présenté ici des applications qu'il est beaucoup plus difficile de concevoir sans recours au traitement d'images.

Deuxième partie

Analyse du contenu de la scène

Chapitre 4

Etat de l'art et architecture proposée pour la détection de personnes

Sommaire

4.1 État de l'art de la détection de personnes 76
4.1.1 Systèmes de détection de personnes 76
4.1.1.1 Détection de visages 77
4.1.1.2 Détection de personnes 79
4.1.1.3 Conclusion sur les méthodes existantes de détection de personnes 80
4.1.2 Modèles d'arrière-plan pour l'extraction des objets d'intérêt . . . 81
4.1.2.1 Méthodes prédictives 81
4.1.2.2 Méthodes non-prédictives 82
4.1.2.3 Conclusions sur les modèles d'arrière-plan existants . . 83
4.2 Architecture générale du système proposé 83
4.2.1 Cartes de probabilités 85
4.2.1.1 Définition 86
4.2.1.2 Caractéristiques de bas-niveau à extraire 86
4.2.1.3 Combinaisons entre cartes de probabilités 87
4.2.2 Extensions du schéma général 88
4.2.2.1 Suivi de personnes 88
4.2.2.2 Mise à jour des caractéristiques de bas-niveau 88

La détection de personnes a toujours été un problème central dans le domaine de l'analyse d'image. La première raison de l'engouement général pour la reconnaissance de la forme humaine provient du nombre important d'applications qui peuvent découler d'un détecteur de personnes performant. Sans même se limiter à la vidéosurveillance, pour laquelle la détection de personne est bien évidemment un outil majeur, les domaines applications d'un tel détecteur vont de l'indexation automatique d'archives vidéo aux outils de

post-traitement pour le cinéma, en passant par les applications militaires. Une autre raison pour laquelle la détection de personnes est un sujet majeur de l'analyse d'image concerne la complexité de cette tâche. L'apparence humaine telle qu'elle est perçue dans une image est en effet l'une des formes les plus difficiles à décrire, justement à cause de son caractère multiforme. Les poses sont variées, et souvent propres à un contexte précis, à une position de caméra donnée. Les vêtements n'aident pas à résoudre le problème, en apportant encore plus de variété en terme de couleurs et de textures possibles. Les approches existantes de détection de personnes imposent toujours des contraintes fortes aux séquences vidéos à traiter, et fonctionnent généralement pour des types de contextes très spécifiques. L'Homme possède en apparence une multitude de caractéristiques qui nous permettent de pouvoir le reconnaître dans une image. Ces caractéristiques concernent différents aspects du corps et de son comportement, comme la peau [SC00], la silhouette [AT06], le mouvement [LG02], le visage [VJ04],[RBK98]. La difficulté avouée du problème de détection de personnes en fait un des défis les plus prisés de l'analyse d'images.

Dans ce chapitre, un bref aperçu des méthodes de détection de personnes existantes les plus importantes est donné. Nous nous intéresserons particulièrement aux détecteurs de visages ainsi qu'aux méthodes permettant la détection du corps humain globalement. Par la suite, un état de l'art des méthodes de soustraction de fond sera établi. Ce problème est en effet un élément important de nombreux détecteurs de personnes, car il permet d'extraire les silhouettes dans une séquence vidéo.

4.1 État de l'art de la détection de personnes

4.1.1 Systèmes de détection de personnes

La détection de personnes dans un environnement de véhicule de transport en commun est un problème très spécifique, avec son lot de contraintes et d'hypothèses particulier. Il s'agit d'un sujet qui n'a pas été traité par le passé, mais pour lequel certaines méthodes existantes, développées pour des contextes différents, méritent d'être évaluées. C'est aussi un problème très vaste qui recoupent plusieurs thèmes plus génériques, comme la détection de mouvement, l'extraction de caractéristiques ou l'estimation robuste de paramètres.

Lorsqu'il s'agit de concevoir un algorithme de détection de personnes, les contraintes imposées par le contexte doivent être analysées car elle permettent de guider le choix des caractéristiques du corps humain à prendre en compte. Une application qui a été beaucoup étudiée dans des travaux précédents est la détection de piétons, généralement vus depuis une caméra de surveillance installée dans la rue. Dans un tel contexte, l'apparence des personnes à l'image est relativement précise. En ce qui concerne la pose, elles apparaissent debout, et à une distance de la caméra telle qu'elles sont visibles entièrement de la tête aux pieds . L'arrière-plan est quant-à lui considéré être relativement statique. Ce contexte de piétons se rapproche beaucoup de notre contexte de véhicules de transport en commun, en terme d'apparence des personnes à détecter, ce qui nous incite à étudier les différentes méthodes déjà existantes.

La détection de visages est aussi un thème qui a été largement exploré par différents auteurs, mais qui reste toujours d'actualité. Nos passagers apparaissent de face et de profil lors de la montée dans le véhicule, et le visage est généralement visible la majeur partie du temps. Les approches de détection de personnes basées sur la détection du visage sont donc intéressantes pour notre application.

D'autres approches, souvent plus complexes, ne sont pas adaptées à notre contexte, et ne seront donc pas présentées ici. Il s'agit par exemple des méthodes basées sur l'estimation des paramètres d'un modèle 3D de personne par rapport à une image. Elles permettent de détecter une personne ainsi que d'estimer sa posture, et sont plutôt adaptées pour des applications de réalité virtuelle, pour lesquelles on souhaite extraire des informations précises sur l'apparence de la personne. De même, les approches basées sur une vision binoculaire de la scène ne peuvent pas être appliquées ici, de part les contraintes matérielles qui nous sont imposées.

4.1.1.1 Détection de visages

Le visage est une partie du corps humain qui a reçu une attention particulière de la part des chercheurs en image. Ceci est dû au fait que les applications basées sur une détection et une bonne localisation des visages dans une image ne manquent pas. On pensera tout d'abord à la biométrie, qui est un domaine où la reconnaissance des visages prend une place importante. Certains systèmes d'aide à la conduite automobile surveillent le visage du conducteur et son expression faciale afin de prévenir tout risque qu'il s'endorme. Dans ce type d'application, la détection de visage est aussi un élément important. Enfin, la plupart des systèmes nécessitant un module de reconnaissance de visages intègrent aussi un module de détection et de localisation des visages. Cela concerne évidemment les système de surveillance vidéo, mais aussi les outils d'indexation automatique d'archives de photos ou vidéos.

La détection de visages est un problème difficile, principalement à cause de la grande variété en apparence qu'il faut prendre en compte.

- Les visages peuvent tout d'abord apparaître sous différents angles, par exemple de face, de trois-quart ou de profil.
- L'apparence d'un visage est aussi spécifique à chaque individu, en terme de forme, de taille ou de couleur et peut comporter ou non des éléments comme une barbe ou des lunettes.
- Une même personne peut avoir des expressions faciales différentes, ce qui contribue encore à la variabilité en apparence.
- Un visage peut aussi être occulté en partie, par un autre objet de la scène.
- Enfin les conditions d'acquisition de l'image peuvent varier en fonction du matériel utilisé et de l'environnement. Cela provoque des différences de couleur et de luminosité qu'il faut prendre en compte.

Les différentes méthodes de détection de visages peuvent être classées en plusieurs types d'approches :

Les **méthodes basées sur des règles** définissant l'apparence d'un visage, à partir de la connaissance humaine de ce qui compose typiquement un visage. Généralement, ces règles représentent les relations spatiales entre les différentes caractéristiques du visage (généralement des régions avec une intensité donnée). Elles sont définies manuellement par un expert, et ne doivent pas être trop détaillées sous peine de perdre en généralité pour la détection. De même les règles ne doivent pas non plus être trop générales pour éviter les fausses détections intempestives. [YH94] a proposé un tel détecteur fonctionnant en multirésolution sur 3 niveaux hiérarchiques de règles, afin de sélectionner rapidement les zones de l'image comprenant probablement un visage. Les règles utilisées imposent par exemple des différences importantes de niveaux de gris entre régions voisines, ou encore des régions dont l'intensité est quasiment uniforme. La robustesse aux variations de pose est limitée, car elle imposerait de définir un jeu de règles pour chacune des nombreuses poses attendues.

Les **approches basées sur l'extraction de caractéristiques invariantes** cherchent à définir un visage comme un ensemble de structures qui existent quelque soit la pose, l'éclairage, ou l'expression. Elles se basent principalement sur la recherche d'éléments composant un visage tels que les yeux, les sourcils et la bouche. [LBP95] a développé une méthode pour localiser un visage dans une scène complexe basée sur la détection de caractéristiques locales et d'association de graphes aléatoires. Les caractéristiques locales (yeux, sourcils et jonction nez/lèvre) sont localisées sur des images d'apprentissage et leurs distances relatives sont modélisées par un mélange de Gaussiennes. La détection d'un visage dans une image de test est alors formulée comme une recherche de graphe dont les nœuds sont les caractéristiques faciales détectées, et les arêtes sont les distances entre caractéristiques. D'autres méthodes utilisent principalement la teinte chair, qui est reconnue comme une caractéristique efficace pour la détection de visages et de mains. Des études ont montré que la variété de la couleur de la peau parmi les différents individus s'exprime beaucoup plus en terme d'intensité que de chrominance [GCPC95] [GCG+96]. Différents espaces couleurs ont été utilisés pour la détection de la peau, parmi lesquels RGB [SSS+96], RGB-normalisé [CBBS94] [SHW98] [QSM98] [SC00], HSV [SF96] ou encore YCrCb [CN98]. D'autre part, [THC03] détecte les visages par un modèle de teinte chair basé sur une combinaison des techniques d'histogramme et de data-mining. La peau est une caractéristique efficace à condition que le contexte permette au détecteur de s'adapter aux différents changements d'illumination de la scène qui peuvent survenir. [MRG98] a présenté un modèle de teinte chair s'adaptant aux variations d'illumination.

Les **méthodes de *template matching*** utilisent plusieurs images d'exemple de visages comme description de l'apparence d'un visage, avec différentes variations en terme de point de vue. En corrélant une image en entrée avec les exemples de visages, il est possible de localiser les visages dans l'image d'entrée. Les exemples de visages peuvent être un ensemble fixe de visages, appris par exemple par ACP sur une base de données [SI95] ou un modèle déformable [KL94] qui permet de prendre en compte une plus grande variabilité.

D'autres auteurs [SO95] cherchent à résoudre le manque de généralisation des ensembles fixes de visages d'exemple en considérant plutôt des sous-parties du visage. La détection d'un visage est alors obtenue à partir de la présence possible des sous-parties dans l'image, déterminée selon la théorie de Dempster-Shafer.

Enfin, les **méthodes basées sur l'apparence** apprennent un modèle de visage à partir d'un ensemble d'exemples, qui représentent le plus fidèlement possible les variations en pose, en illumination ou en expression qui peuvent être attendues. La détection s'effectue alors à partir du modèle appris. Le problème de détection est souvent formulé de manière statistique comme une classification Bayesienne entre deux classes visage et non-visage de densités $p(\boldsymbol{x}/visage)$ et $p(\boldsymbol{x}/non-visage)$, avec $\boldsymbol{x}$ l'hypothèse d'un visage, comprenant en particulier les paramètres de localisation dans l'image. La manière la plus classique de construire un modèle de visage à partir d'images d'exemple est d'effectuer une analyse en composantes principales (ACP) sur l'ensemble des images d'exemples. Il a été montré par [KS90] que les images de visages peuvent être encodées à partir d'un nombre réduit d'images de bases. Les vecteurs de base issue de la ACP sont appelés *eigenfaces*. Ces bases d'images ont été appliquées pour la détection de visage par [PMS94], et plus récemment comparées à des sous-espaces non-linéaires dans [SM04]. [RBK98] présente un système de détection de visages vus de face basé sur des réseaux de neurones, entraînés grâce à un algorithme de type *bootstrap*. Un algorithme de détection de visage fonctionnant en temps-réel a été proposé par [VJ04]. Il est basé sur la construction d'un ensemble de classificateurs en cascade, sélectionnés par l'algorithme AdaBoost de [FS95]. Les caractéristiques visuelles sélectionnées pour la classification sont simplement des différences de sommes d'intensités entre des rectangles connexes, ce qui permet un calcul rapide.

4.1.1.2 Détection de personnes

La détection de personnes dans les images a aussi été étudiée pour les cas où les individus apparaissent de plus loin. On retrouve ce genre de situations dans les applications où des piétons doivent être détectés et suivis, typiquement en vidéosurveillance de scènes de rues. Dans ce type de contexte, de part la distance et l'orientation du sujet par rapport à la caméra qui ne permet souvent pas une très bonne visibilité du visage, les techniques employées sont différentes de celles présentées précédemment. La détection de piétons est une discipline qui se rapproche beaucoup de notre contexte de véhicules de transport en commun, l'apparence des personnes à l'image étant similaire. Les difficultés les plus importantes dans l'élaboration d'une méthode de détection de personnes sont la modélisation correcte du corps humain avec toute sa variabilité en apparence, et la prise en compte des occultations partielles. Nous présentons donc ici les principales méthodes existantes pour la détection de piétons.

Les premières approches pour la modélisation et la détection de piétons utilisent des modèles représentant le corps humain dans sa globalité. [OPS+97] propose un système qui apprend à partir d'exemples un modèle d'apparence du corps humain. La représentation des images par leurs transformées en ondelettes suivant une base de fonctions redondantes

permet d'extraire des propriétés invariantes aux changements en couleur et en texture. Les méthodes développées par [Gav00] et [Fel01] utilisent quant-à-elles les silhouettes de personnes pour l'apprentissage et la détection. Tandis que [Fel01] se base sur la distance de Hausdorff pour déterminer la similarité entre une forme extraite par une détection de contours et un modèle de personne appris à partir d'exemples, [Gav00] travaille sur des transformées de distances à partir du contour des objets. D'autres auteurs proposent de détecter des caractéristiques locales du corps indépendamment les unes des autres, plutôt que de construire un modèle d'apparence globale. Ainsi, le système proposé par [VJS03] apprend une représentation de l'apparence humaine à partir de caractéristiques de différences d'intensités entre rectangles adjacents, sélectionnées automatiquement par l'algorithme AdaBoost pour une classification efficace. [MPP01] modélise plus explicitement des parties reconnaissables du corps humain, à savoir la tête, les jambes et les deux bras, grâce à des machines à support de vecteurs (SVM) classifiant les coefficients de la transformée en ondelettes de Haar sur des exemples positifs et négatifs. La détection de personne est réalisée à partir des parties détectées par une autre classification de type SVM. [MSZ04] détecte aussi différentes parties du corps séparément, reliées géométriquement par des densités de probabilité jointes. Les caractéristiques utilisées dans la modélisation des parties du corps sont basées sur l'orientation des contours. Globalement, la robustesse aux occultations partielles est plus forte pour les modèles représentant des parties du corps séparément que pour les modèles d'apparence globale. Plus récemment, [LSS05] propose un algorithme combinant des caractéristiques locales et globales, provenant de sources différentes, par une segmentation probabiliste.

4.1.1.3 Conclusion sur les méthodes existantes de détection de personnes

La détection de personnes dans une image est donc loin d'être un problème nouveau. Les essais sont nombreux, avec des résultats plus ou moins convaincants selon les auteurs. Comme il a été dit précédemment, il n'existe pas de méthode universelle pour la détection de personnes, qui puisse fonctionner dans la majorité des cas. Chaque contexte apporte ses particularités qu'il faut prendre en compte. Les méthodes existantes qui viennent d'être présentées n'échappent pas à cette règle. Notre contexte de vidéosurveillance embarquée est particulièrement contraignant, que ce soit du point de vue des perturbations en illumination et en mouvement ou de la qualité de la vidéo imposée par le matériel. Ainsi, les méthodes basées sur la reconnaissance de la structure générale du visage sont difficilement applicables à cause des fortes variations de pose et de la qualité médiocre des images acquises. En particulier, les méthodes de type *eigenfaces* demanderaient une base d'apprentissage suffisamment complète pour englober tous les cas possibles. Les détecteurs de personnes qui prennent en compte le corps humain dans sa globalité sont déjà plus intéressants pour notre contexte. Nous travaillons sur des séquences vidéo, ce qui nous permet d'envisager l'extraction des silhouettes des passagers. C'est une information qu'il ne faut pas négliger, malgré les nombreuses occultations partielles des personnes auxquelles il faut s'attendre. Les modèles complets de personnes, représentant le corps en multiples parties, ne sont d'ailleurs pas indispensables, car nous nous limitons bien aux problèmes

de détection et de suivi de personnes, et non à l'estimation de la posture.

En définitif, sans ignorer les diverses méthodes existantes, nous allons définir un détecteur spécifique aux particularités de notre contexte qui, sans se prétendre supérieur à l'existant, sera bien adapté à nos conditions. La stratégie employée pour la résolution de notre problème consiste à combiner des informations bas-niveau extraites des séquences vidéo dans un modèle de personne suffisamment général pour satisfaire la grande variabilité de pose, mais assez spécifique pour limiter les fausses détections. L'extraction des informations bas-niveau de teinte chair et d'avant-plan est un problème à part entière, car elle doit être fait elle aussi de manière robuste aux perturbations de la vidéo. Aussi, nous verrons comment la détection simultanée de plusieurs personnes est rendue possible par le cadre probabiliste défini.

4.1.2 Modèles d'arrière-plan pour l'extraction des objets d'intérêt

L'extraction des pixels d'avant-plan dans une séquence vidéo est un problème qui nous intéresse particulièrement pour l'élaboration de notre détecteur de personnes. Un algorithme permettant de détecter efficacement les pixels d'avant-plan peut effectivement servir à extraire les silhouettes des passagers. L'avant-plan d'une scène filmée peut être défini comme l'ensemble des éléments qui ont un comportement en mouvement particulier par rapport à l'arrière-plan. Dans le cas classique d'une caméra statique, la détection des pixels d'avant-plan revient donc à chercher dans chaque image les pixels qui appartiennent à un objet qui est, ou a été, en mouvement. De ce fait, l'extraction des pixels d'avant-plan est souvent reliée à l'estimation de l'arrière-plan de la scène, dont le comportement peut être modélisé plus aisément. Cet arrière-plan est en effet souvent considéré comme statique, ou possédant des caractéristiques permettant d'estimer ses transformations au cours du temps. A partir d'une estimation de l'arrière-plan, l'avant-plan est simplement considéré comme l'ensemble des pixels qui n'appartiennent pas au modèle d'arrière-plan. Les méthodes de détection d'avant-plan utilisant ce principe sont communément regroupées sous le terme soustraction de fond. Un des premiers travaux dans ce domaine, proposé par [LY87], consistait simplement à soustraire explicitement l'image courante à une image d'arrière-plan fixe, puis d'appliquer un seuil sur le résultat pour détecter les objets d'avant-plan. Il est évident que cette méthode ne peut pas être efficace en présence d'un arrière-plan dynamique. La majorité des travaux ultérieurs dans ce domaine se sont donc focalisés sur le développement de méthodes de soustraction de fond fonctionnant lorsque l'arrière-plan comporte des variations.

Les algorithmes d'estimation d'arrière-plan peuvent être classées en deux familles, que l'on appellera méthodes prédictives et méthodes non-prédictives.

4.1.2.1 Méthodes prédictives

La particularité des méthodes prédictives est de se baser sur des estimations passées de l'arrière-plan et un modèle dynamique afin de prédire l'état attendu de l'arrière-plan à un instant donné. L'image observée à cet instant est alors comparée à la prédiction.

Pour chaque point de l'image on mesure la déviation par rapport à la prédiction, qui peut ensuite être utilisée pour détecter les points d'avant-plan et pour mettre à jour l'estimation de l'arrière-plan. Ce mécanisme confrontant la prédiction et l'observation rappelle le principe du filtrage de Kalman, qui a d'ailleurs été utilisé directement pour l'estimation d'arrière-plan par plusieurs auteurs [WHKG90] [KWM94]. Une version simplifiée du filtrage de Kalman, connue sous le nom de filtrage de Wiener est aussi utilisée dans le cadre de la détection de personnes par [TKBM99]. Les données utilisées pour la modélisation du fond ne sont pas toujours directement les valeurs des pixels. Il est souvent plus approprié de convertir les données dans un espace où les variations de l'arrière-plan sont plus facilement contrôlables. [ZS03] travaille par exemple dans un espace obtenu par analyse en composantes principales, et modélise les variations de l'arrière-plan par un processus autorégressif. [MMPR03] propose une technique similaire pour l'estimation et la soustraction de fond, aussi basée sur l'idée originale de [DCWS03], consistant à considérer la séquence vidéo comme une série temporelle évoluant suivant un modèle autorégressif, dont l'estimation des paramètres permet de prédire l'image suivante. Plus récemment, [XJ04] propose de discriminer les régions d'avant-plan des régions d'arrière-plan en introduisant un ensemble de filtres adaptatifs capables de modéliser les changements spatiaux locaux. Les valeurs de corrélation de ces filtres sont utilisées pour discriminer les deux types de régions. Cette méthode est efficace pour les scènes en extérieur comportant des petits mouvements provenant d'arbres ou de plans d'eau.

4.1.2.2 Méthodes non-prédictives

Les méthodes non-prédictives constituent l'autre famille d'algorithmes d'estimation d'arrière-plan. Le principe commun à ces méthodes est de modéliser l'ensemble des états possibles de l'arrière-plan sans tenir compte de l'ordre temporel des images observées. Il s'agit donc de modéliser les variations de l'arrière-plan, puis de détecter les pixels d'avant-plan comme ceux qui ne correspondent pas à ce modèle. Le modèle est souvent interprété de manière probabiliste, et l'on cherche à estimer la distribution des pixels de l'arrière-plan. [WADP97] a commencé par modéliser chaque pixel de l'image et ses variations par une distribution Gaussienne. [FR97] a proposé ensuite d'utiliser plutôt un mélange de trois Gaussiennes afin de prendre en compte la multimodalité de la distribution des valeurs d'un pixel que l'on trouve dans la plupart des applications. Dans son contexte de surveillance de trafic routier, chaque Gaussienne modélise l'une des trois classes considérées que sont la route, les ombres, et les véhicules. L'estimation des paramètres du modèle est réalisée par l'algorithme *Expectation-Maximization* (EM). Le système proposé peu après par [GS99] est une extension de cette idée à un nombre quelconque de Gaussiennes. Une approximation rapide de l'algorithme EM est aussi développée pour estimer les paramètres du modèle en temps réel. La modélisation des variations de la scène par des mélanges de Gaussiennes est une méthode qui est devenue populaire, et est souvent utilisée comme méthode de base pour le développement de méthodes plus complexes. Par exemple, [Har02] propose de guider la mise à jour des paramètres du mélange de Gaussiennes à l'aide de modules de plus haut-niveau, opérant par rétroaction. Le type de modèle utilisé pour caractériser la

distribution des états d'un pixel n'est pas toujours basé sur un mélange de Gaussiennes. Les variations sont parfois trop complexes pour être modélisées de la sorte, et une approche non-paramétrique peut alors convenir. Par exemple [EHD00] propose une telle approche en utilisant des noyaux Gaussiens sur un nombre réduit d'observations pour prendre en compte les changements brusques de l'arrière-plan. [MP04] modélise les aspects dynamiques de l'arrière-plan en combinant des informations de mouvement obtenues par l'estimation du flot optique, et l'information sur l'intensité des pixels dans un même vecteur de caractéristiques. Sa méthode d'estimation de l'arrière-plan allie donc l'information spatiale à l'information temporelle. L'utilisation de noyaux Gaussiens dont la bande passante est variable pour chaque observation en fonction de l'incertitude des informations extraites donne de bons résultats pour des contextes difficiles.

4.1.2.3 Conclusions sur les modèles d'arrière-plan existants

Les changements forts et brusques d'illumination observables dans les séquences vidéo de notre contexte nous font considérer en priorité les méthodes d'estimation prédictives, considérant l'arrière-plan comme une série temporelle dont le comportement est modélisable à partir des observations passées. L'hypothèse d'arrière-plan statique, ou quasiment statique, adoptée par la grande majorité des méthodes existantes, n'est pas valable dans notre contexte dans le cas général, à cause du mouvement apparent du paysage extérieur, visible dans les parties vitrées. Néanmoins nous nous intéresserons dans la suite au cas plus simple des séquences d'arrêt du véhicule, pour lesquelles l'arrière-plan est considéré quasiment statique, avec des légers mouvements locaux. Ces séquences possèdent des petits mouvements, dus à la végétation ou simplement au déplacement relatif du paysage par rapport au véhicule provoqué par ses suspensions, qu'on ne peut pas ignorer pour l'estimation de l'arrière-plan. Les mouvements locaux et changements d'illumination de nos séquences sont *a priori* très difficile à prédire, car ils ne correspondent pas à des changements à long terme, ou périodiques, comme il est supposé implicitement dans les méthodes [DCWS03], [MMPR03], [ZS03], basées sur une modélisation autorégressive d'évolution de l'arrière-plan. Aussi, les méthodes non-prédictives basées sur les mélanges de Gaussiennes ont l'avantage de modéliser plusieurs états possibles pour une position de l'image donnée. La durée très courte des séquences d'arrêt, et le nombre important de passager occultant l'arrière-plan nous incitent à considérer plusieurs hypothèses de fond pour l'arrière-plan. L'utilisation d'hypothèses multiples est un principe qui se rapproche des Gaussiennes des méthodes non-prédictives. La méthode que nous allons développer pour l'estimation d'arrière-plan à un arrêt présentera donc des aspects tirés des méthodes prédictives, combinés à des aspects provenant des méthodes non-prédictives.

4.2 Architecture générale du système proposé

Le contexte très particulier de la vidéosurveillance embarquée nous incite à développer un système de détection de personnes adapté aux nombreuses contraintes qui sont imposées.

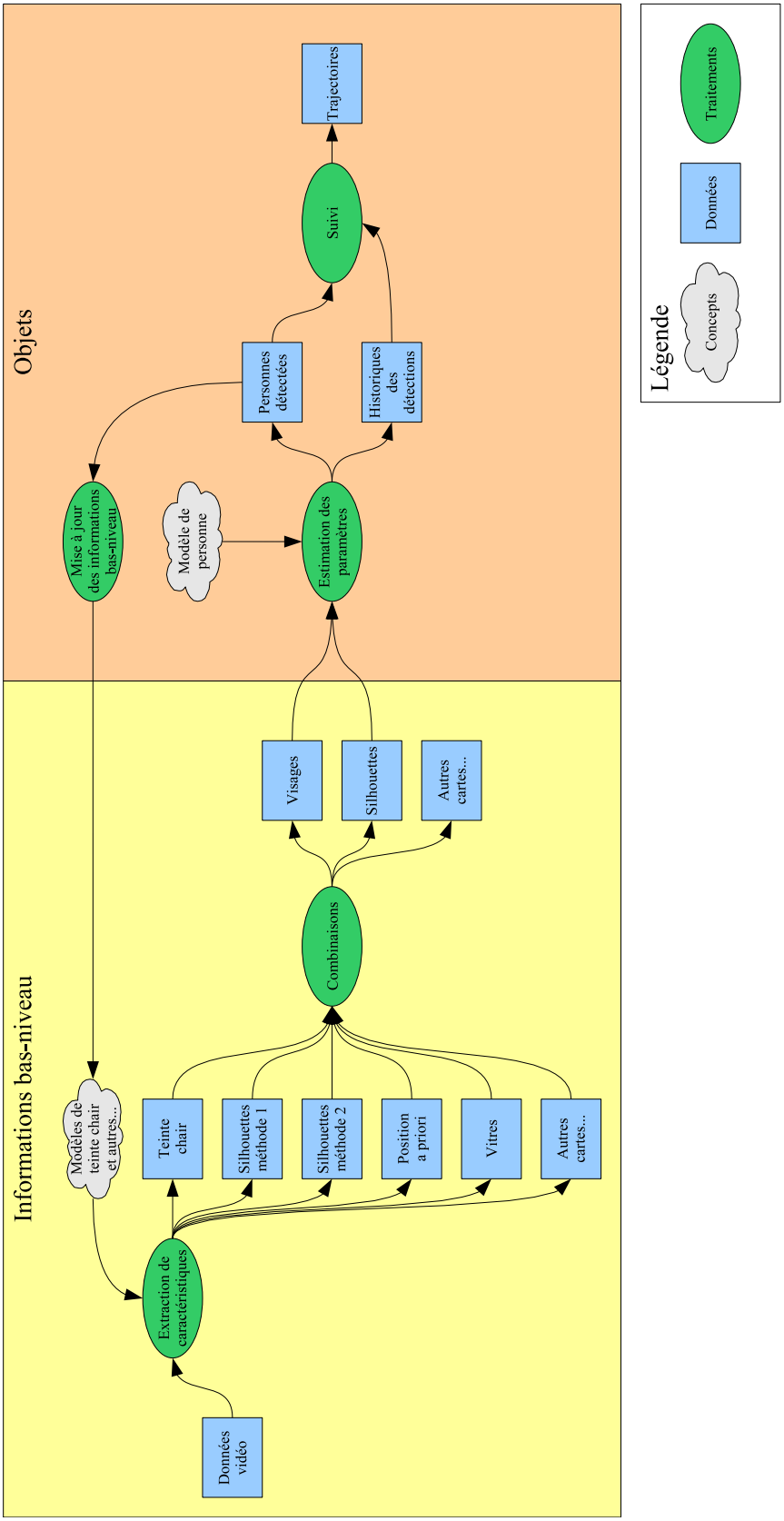


FIG. 4.1 – Schéma global du détecteur de personnes

Le principe général du système proposé est décrit ici. La figure 4.1 schématise les différents éléments constituant le détecteur de personnes. Il est décomposé en deux grandes étapes successives :

1. L'extraction d'informations bas-niveau et leur représentation sous forme de cartes de probabilités
2. Le passage au niveau objet, avec la définition d'un modèle de personne et l'estimation de ses paramètres.

Ces deux étapes vont être l'objet principal des études menées. Les deux prochains chapitres sont d'ailleurs consacrés aux solutions envisagées pour chacune.

La conception d'un système de détection de personnes est un travail qui demande de tester plusieurs solutions pour chacun des sous-problèmes auquels on est confronté. Ainsi l'architecture du système a été pensée de façon très modulaire, afin qu'il soit possible de remplacer un élément de la chaîne de traitement par un autre sans conséquences sur les autres éléments. Cette modularité se retrouve premièrement dans la séparation du détecteur en les deux grandes étapes citées plus haut, avec la possibilité de considérer plusieurs modèles de personnes fonctionnant sur les mêmes informations bas-niveau. Aussi, il est possible d'envisager différents algorithmes pour l'extraction des informations bas-niveau, voire des algorithmes qui n'extraient pas exactement la même information, mais dont le résultat est proche sémantiquement. Par exemple, l'information de mouvement peut être interchangée avec une autre information sémantiquement proche, comme une détection des pixels d'avant-plan. La représentation des informations extraites sous forme de carte de probabilités est aussi une caractéristique fondamentale de notre système. Cela permet en effet de travailler avec les différentes informations de manière unifiée. Il est alors possible de fusionner des sources d'information afin d'en créer une troisième. Ainsi la teinte chair peut être combinée avec l'information d'avant-plan pour obtenir une carte de probabilités de peau, robuste aux éléments fixes de la scène qui ne sont pas des objets d'intérêt mais qui pourraient avoir une teinte similaire à la peau. Les probabilités permettent aussi de combiner les sources d'information sans prendre de décision prématurée.

4.2.1 Cartes de probabilités

La détection des personnes dans nos séquences vidéo commence par une analyse bas-niveau des données, en vue de l'extraction de certaines caractéristiques pour chaque pixel. Par bas-niveau, on entend principalement que les informations que nous souhaitons extraire portent sur chaque pixel indépendamment. Il n'y a donc pas de notion d'objet à cette étape du détecteur. Grâce à une représentation des caractéristiques extraites sous forme de cartes de probabilités, deux avantages apparaissent clairement :

1. Tout d'abord, les caractéristiques sont représentées de manière unifiée, quelque soit leur nature, sous une forme qui permet de les combiner simplement. Deux méthodes permettant d'extraire les pixels du visage des personnes peuvent alors résulter en des cartes de probabilités très similaires, qu'il est possible d'interchanger ou de combiner. Par exemple, une méthode basée sur la teinte chair et une méthode basée sur la

reconnaissance de forme ont des principes bien différents, mais les cartes de probabilités extraites seraient similaire sémantiquement, car elles permettraient toutes deux l'extraction des pixels des visages.

2. L'autre avantage d'une représentation sous forme de carte de probabilités est la possibilité de conserver l'incertitude sur l'appartenance d'un pixel à une classe. Par exemple, la teinte chair est une caractéristique couleur relativement floue, et un pixel peut avoir une couleur dont l'appartenance à la teinte chair n'est pas certaine.

4.2.1.1 Définition

Une carte de probabilités est définie pour une image I de dimensions $W \times H$ et pour une classe d'objets C, comme un espace de mêmes dimensions que l'image où chaque pixel $\boldsymbol{k}$ de l'image est associé à sa probabilité $p(\boldsymbol{k} \in C)$ d'appartenance à C , notée aussi $p_C(\boldsymbol{k})$.

Cette définition est très générale et ne suppose pas par exemple l'indépendance de chaque valeur de la carte. Il est d'ailleurs envisagé d'appliquer des opérateurs morphologiques de type dilatation / érosion sur les cartes de probabilités afin d'améliorer la détection, ce qui impose justement une forte dépendance entre pixels voisins.

Le passage du résultat de détection d'une caractéristique à une probabilité est aussi laissé libre, et dépend de la classe C considérée. On définira généralement une densité de probabilité à partir d'un état idéal et des variations admises pour qu'un pixel appartienne à la classe. Parfois la densité de probabilité n'a pas besoin d'être définie explicitement, mais la probabilité de chaque pixel est déduite naturellement des données vidéo. C'est le cas de la carte de probabilité d'appartenance à une région vitrée, comme nous le verrons dans le chapitre suivant.

4.2.1.2 Caractéristiques de bas-niveau à extraire

Les caractéristiques de bas-niveau à extraire doivent être pertinentes pour la description d'une personne. Elles doivent permettre de détecter une partie d'une personne efficacement, sans être trop sensibles aux fausses détections. Pour notre contexte de vidéosurveillance embarquée, nous avons choisi d'étudier principalement deux types de caractéristiques, correspondant l'une au corps dans sa globalité, et l'autre à la tête.

Étant donné les difficultés liées à la détection des visages par la forme dans un environnement contraignant, la caractéristique correspondant à la tête s'est vite limitée à la teinte chair. Le modèle de teinte chair est alors la principale variable qui a été étudiée. La détection des pixels du corps est un problème qui a reçu plus d'attention, avec différents essais de mesures sur l'image, basées sur le mouvement. Seules les mesures les plus concluantes seront présentées dans la suite.

Bien que non directement pertinente pour la description d'une personne, nous considérons l'extraction des pixels appartenant aux vitres. Cela provient à l'origine du constat que le comportement temporel des pixels d'une séquence vidéo typique de notre contexte est très différent pour les régions vitrées et les régions non-vitrées. Les variations dans les régions vitrées sont trop fortes pour que les méthodes classiques de détection de mouvement puisse

extraire la silhouette des personnes correctement. Ainsi une carte de probabilités de vitres peut permettre de séparer l'analyse de la séquence en deux régions, ou du moins de mesurer la confiance que l'on peut porter aux résultats d'un algorithme classique d'extraction de silhouette. En réalité, nous verrons que les études menées sur la détection des personnes se sont rapidement focalisées sur les séquences d'arrêt, où la différence entre les régions vitrées et non-vitrées est moindre (bien que toujours existante). Néanmoins, la carte de probabilités d'appartenance à une région vitrée nous sera utile pour d'autres applications que la détection de personne, comme la localisation automatique des portes ou la compression sélective de la vidéo.

D'autres cartes de probabilité peuvent être considérées pour faire évoluer le détecteur de personnes. Nous avons en particulier envisagé de représenter de l'information *a priori* directement sous la forme de carte de probabilités. Il est par exemple supposé que les visages apparaissent dans une région de l'image spécifique dans notre contexte. Les visages n'apparaissent par exemple jamais dans le bas de l'image. Une carte de probabilité *a priori* de la position des visages peut rendre compte de cette information.

En résumé, les cartes de probabilités que nous avons à disposition sont :

- une carte de probabilités de teinte chair
- une carte de probabilités d'appartenance à la silhouette d'une personne
- une carte de probabilités d'appartenance à une région vitrée
- une carte de probabilités *a priori* de position des visages

Les algorithmes permettant l'obtention de ces cartes sont l'objet du chapitre suivant.

4.2.1.3 Combinaisons entre cartes de probabilités

Les cartes de probabilités extraites des données vidéos sont tirés de caractéristiques simples, comme la couleur et le mouvement. Pour détecter plus efficacement les pixels d'objets de la scène que nous considérons, il est parfois nécessaire de les combiner. Par exemple la teinte chair est une information permettant de détecter les visages, mais elle n'est évidemment pas robuste aux autres objets de la scène dont la teinte est similaire. La combinaison de la teinte chair avec une carte de probabilité représentant le corps des personnes à l'image permet déjà de limiter les fausses détection aux pixels de teinte chair présents sur les personnes (principalement les bras et les jambes). Une autre combinaison avec une carte de probabilité *a priori* de position du visage va pouvoir aussi atténuer les fausses détections de bras et de jambes.

Il est aussi intéressants de combiner des cartes de probabilités provenant de méthodes permettant d'extraire les mêmes éléments de la scène. Pour la détection des pixels de silhouettes, il est par exemple possible de combiner l'information de mouvement inter-image avec une information obtenue par modélisation du fond, afin d'obtenir une carte de probabilités plus robuste.

La combinaison de deux cartes de probabilités p_A et p_B est simplement un processus qui génère une nouvelle carte p_{AB} à partir de ces deux cartes. La façon la plus simple de combiner deux cartes est bien sûr de réaliser la multiplication des deux probabilités pour

chaque pixel :

$$p_{AB}(\boldsymbol{k}) = p_A(\boldsymbol{k}).p_B(\boldsymbol{k}) \tag{4.1}$$

Cette multiplication est apparue suffisante pour les cartes de probabilités que nous souhaitions combiner. Une combinaison plus performante est en fait réalisée en aval pendant la seconde étape du détecteur de personne, grâce au modèle de personne défini à partir de plusieurs sources d'information.

4.2.2 Extensions du schéma général

Sur le schéma général présenté figure 4.1, on note certaines extensions dont nous n'avons pas encore parlé. En particulier on remarquera la présence d'un module de suivi de personnes, fonctionnant à partir du résultat de la détection, ainsi qu'une mise à jour des informations de bas-niveau. Cette mise à jour est contrôlée elle aussi par le résultat de la détection de personne.

4.2.2.1 Suivi de personnes

Le suivi de personnes est l'étape qui succède naturellement la détection de personnes. Nous présenterons dans cette partie, à la fin du chapitre 6, deux méthodes de suivis de personnes. Nous nous sommes beaucoup plus attardé sur le problème de la détection des personnes que sur leur suivi. Néanmoins, les résultats préliminaires obtenus pour le suivi seront présentés car il s'agit d'une extension naturelle du système développé.

4.2.2.2 Mise à jour des caractéristiques de bas-niveau

Comme nous l'avons précisé en introduction, le système de surveillance doit fonctionner de la manière la plus autonome possible, afin de limiter les interventions humaines, et doit s'adapter à un grand nombre de situations. Les caractéristiques de bas-niveau que l'on souhaite extraire doivent donc être robustes aux différentes conditions possibles. La carte de probabilité de teinte chair n'est obtenue que par l'intermédiaire d'une modélisation de la teinte chair. Nous verrons que les conditions d'illumination de la scène font varier l'apparence de la peau des individus bien plus que la diversité naturelle de couleur entre les individus. Aussi, le type de caméra joue beaucoup sur la couleur de la peau visible à l'image. Il est donc envisagé de mettre à jour dynamiquement le modèle de teinte chair afin qu'il s'adapte au mieux aux conditions d'illumination. La détection des personnes permet justement d'obtenir simplement la position des pixels de peau dans l'image. A partir d'un modèle de teinte chair assez standard, et du résultat de détection des personnes, il est alors possible d'optimiser le modèle de teinte chair pour améliorer la détection. Cette idée n'a pas été étudiée de manière approfondie dans ces travaux. Nous incluons néanmoins cette extension dans le schéma général du système de détection de personnes pour mettre en valeur son aptitude à évoluer.

Chapitre 5

Extraction d'éléments de la scène

Sommaire

5.1 Détection de la teinte chair 91
5.1.1 Modélisation 92
5.1.1.1 Espace couleur et modèle 92
5.1.1.2 Discussion sur le choix du modèle 94
5.1.1.3 Apprentissage 94
5.1.2 Détection 95
5.1.3 Mesure de la performance de détection 96
5.2 Détection des parties vitrées 96
5.2.1 Quantification de la variation de chaque pixel 99
5.2.1.1 Comparaison de la partie vitrée et de la partie fixe . . . 99
5.2.1.2 Mesure du contenu haute-fréquence 100
5.2.2 Application à la localisation de la porte du véhicule 101
5.2.2.1 Définition du modèle de porte 101
5.2.2.2 Estimation des paramètres par simulation Monte-Carlo 102
5.3 Détection des pixels d'avant-plan par des méthodes classiques 103
5.3.1 Détection du mouvement inter-image 105
5.3.2 Modélisation du fond 107
5.3.2.1 Image moyenne 107
5.3.2.2 Modélisation Gaussienne des variations 108
5.3.2.3 Détection des pixels d'avant-plan 110
5.4 Estimation du fond de la scène à un arrêt 111
5.4.1 Caractérisation des séquences d'arrêt du véhicule 111
5.4.1.1 Variations d'illumination 111
5.4.1.2 Mouvements d'objets 112
5.4.2 Présentation générale de la méthode proposée d'estimation du fond à un arrêt 112
5.4.2.1 Formalisation du problème 112

5.4.2.2 Modélisation de l'arrière-plan 113
5.4.3 Transformation linéaire des pixels d'arrière plan 113
5.4.3.1 Estimation des paramètres de la transformation linéaire 114
5.4.3.2 Transformation optimale au sens des moindres carrés pondérés 115
5.4.3.3 Estimation itérative des poids et de la transformation . 116
5.4.4 Extraction des pixels d'avant-plan à partir de l'arrière plan estimé 117
5.4.4.1 Analyse des résidus 117
5.4.5 Extension aux vecteurs de caractéristiques 119
5.4.5.1 Vecteur du voisinage d'un pixel 120
5.4.5.2 Vecteur de statistiques sur la distribution locale des couleurs 120
5.4.5.3 Pertinence d'une transformation linéaire des moments statistiques 121
5.4.6 Hypothèses multiples sur l'état de l'arrière-plan 122
5.4.6.1 Besoin d'hypothèses multiples 123
5.4.6.2 Lien entre vecteurs de caractéristiques et hypothèses . . 124
5.4.7 Évaluation des performances 125
5.4.7.1 Évaluation de l'estimation de l'arrière plan 126
5.4.7.2 Comparaison quantitative avec les mélanges de Gaussiennes 128
5.4.7.3 Performances sur séquences réelles 128
5.4.8 Implémentation efficace du calcul des vecteurs 132
5.4.8.1 Expression des moments centrés multivariés 132
5.4.8.2 Calcul rapide des sommes de puissances 132
5.4.8.3 Considérations sur l'espace mémoire 133
5.4.9 Conclusions sur l'estimation du fond à un arrêt 133
5.5 Conclusions sur l'extraction d'informations bas-niveau 134

Les principales applications que nous souhaitons réaliser par analyse d'images requièrent l'extraction de certains éléments de la scène. Ces éléments peuvent être issus de caractéristiques bas-niveau de l'image, telle que la couleur pour la détection de teinte chair. D'autres objets que nous souhaitons extraire demandent une analyse plus fine de la séquence vidéo. En particulier la détection de personnes nécessite l'extraction d'éléments plus simples, pour ensuite les combiner. Ce chapitre a pour but de présenter des méthodes d'extraction de certains éléments de la scène qui sont utilisés ensuite comme base pour extraire des objets plus complexes, ou pour construire directement des applications.

Les éléments que nous souhaitons extraire de la scène et qui sont étudiés dans ce chapitre sont les suivants :

– la teinte chair
– les parties vitrées du véhicule
– les objets en mouvement

– l'arrière-plan et l'avant-plan d'une scène d'arrêt

Pour chacun de ces éléments, le résultat de la détection se présente sous la forme d'une carte de probabilités, pour les raisons évoquées dans le chapitre 4, à savoir une combinaison simple des informations et une représentation de l'incertitude.

La détection de la teinte chair est un thème qui a été étudié de manière intensive par de nombreux auteurs en analyse d'image. Nous n'apportons pas ici de contribution notable sur la modélisation ou la détection de la teinte chair par rapport à l'existant, mais proposons plutôt une étude de son applicabilité à notre contexte de vidéosurveillance embarquée. Notamment, nous étudions les effets de la compression JPEG et des contrôles automatiques de gain et de balance des couleurs sur la qualité de la détection des pixels de peau.

Les parties vitrées du véhicule sont un élément de la scène que nous souhaitons extraire. Son étude à été motivée par le fait que le comportement temporel des séquences vidéo considérées est très différent dans les régions vitrées par rapport aux autres régions, ce qui peut nuire au bon fonctionnement des algorithmes de détection de pixels d'avant-plan. L'extraction des régions vitrées permettrait alors de connaître les zones de l'image où la détection de pixels d'avant-plan n'est pas fiable. D'autre part, la détection des régions vitrées est utile à certaines applications envisagées, notamment la compression sélective et la localisation des portes du véhicule. L'algorithme proposé ici est basé sur une étude du comportement temporel de chaque pixel, globalement sur une longue durée. Nous verrons que la puissance moyenne de la dérivée temporelle de l'intensité est une mesure discriminant efficacement les régions vitrées des régions non-vitrées.

En ce qui concerne la détection des objets en mouvement, ce chapitre présente les résultats obtenus par des algorithmes classiques de soustraction inter-images et de soustraction de fond, sur des séquences vidéo où le véhicule est en mouvement. Cette partie du chapitre sert en réalité d'introduction à la modélisation d'arrière-plan à un arrêt. Nous verrons en effet que les perturbations de la scène sont trop importantes lorsque le véhicule est en mouvement pour permettre une extraction satisfaisante des pixels d'avant-plan.

Le reste du chapitre est consacré à l'extraction des pixels d'avant-plan lors d'un arrêt du véhicule. Un modèle d'arrière-plan mettant en œuvre la distribution locale des couleurs autour de chaque pixel, et estimant la transformation globale des couleurs des pixels d'arrière-plan est décrit. L'utilisation de plusieurs hypothèses pour la description de l'arrière-plan permet de prendre en compte les cas ou les objets d'avant-plan créent de fortes occultations. Ce modèle permet l'extraction efficace des pixels d'avant-plan, comme le montrent les résultats sur des séquences vidéo réelles et synthétiques. Une comparaison avec la méthode classique des mélanges de Gaussiennes est réalisée, afin de montrer la supériorité de notre algorithme dans notre contexte de vidéosurveillance embarquée.

5.1 Détection de la teinte chair

La teinte chair est une information classiquement utilisée pour la détection de visage. La peau possède effectivement une plage de teintes très spécifique, de telle manière qu'une

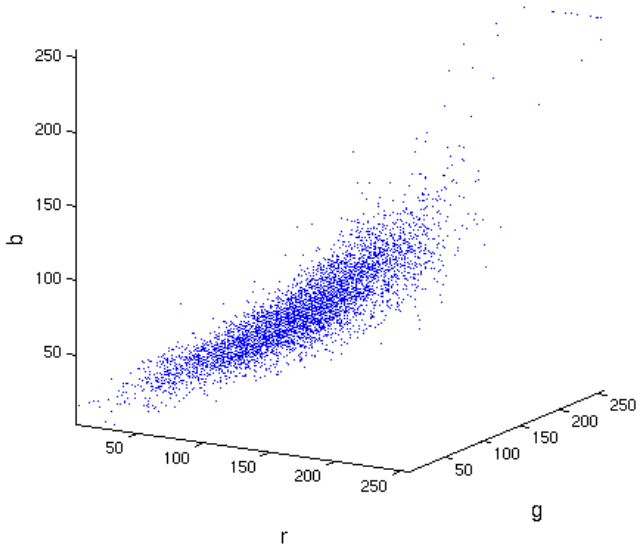


FIG. 5.1 – Répartition de la teinte chair dans l'espace couleur RGB

modélisation correcte de cet ensemble de couleur permet de détecter les pixels de peau dans une image avec un taux de fausses alarmes relativement réduit. Néanmoins, les variations d'illumination et la présence d'objets de couleur similaire rendent la détection de pixels de peau plus difficile, surtout lorsqu'il n'est pas possible de calibrer la caméra en balance des couleurs, et que la scène est soumise à plusieurs types d'éclairage suivant les instants de journée. L'information de teinte chair n'est toutefois pas à délaisser, car son extraction est relativement simple et le résultat obtenu peut être fusionné avec d'autres caractéristiques de l'image, en vue de détecter les personnes.

5.1.1 Modélisation

La détection des pixels de teinte chair dans une image nécessite que l'on sache modéliser correctement l'ensemble des couleurs que peut prendre la peau. Une modélisation statistique, basée sur l'apprentissage des teintes à partir d'exemples, est ici utilisée. Il est nécessaire de se fixer un espace couleur dans lequel travailler, ainsi qu'un modèle capable de représenter la teinte chair dans cet espace. Après avoir estimé les paramètres du modèle à partir de pixels d'exemple, la détection des pixels de peau est rendue possible pour une image quelconque.

5.1.1.1 Espace couleur et modèle

Le choix de l'espace couleur et celui du modèle doivent être fait de manière conjointe. En effet, la teinte chair occupe une région dont la forme varie selon l'espace couleur.

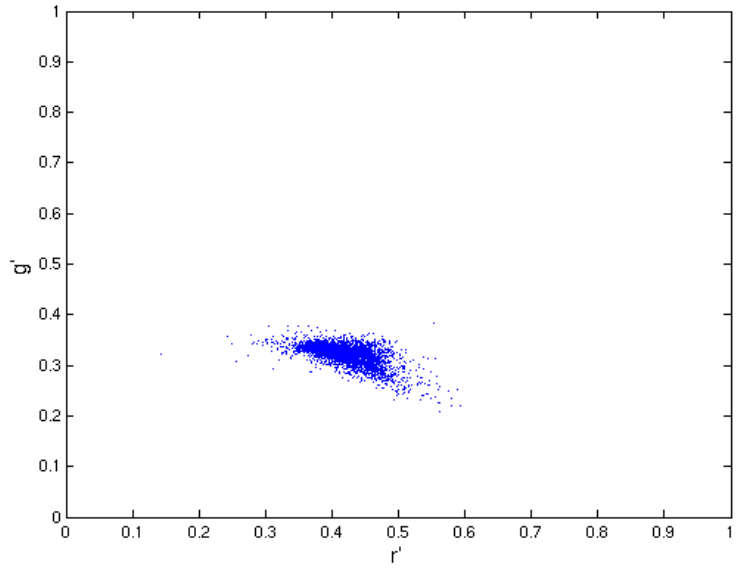


FIG. 5.2 – Répartition de la teinte chair dans l'espace de chrominance rg-normalisé

La figure 5.1 illustre la répartition des pixels de teinte chair dans l'espace RGB. On remarque que cette répartition impose l'utilisation d'un modèle non linéaire, si l'on désire modéliser la teinte chair directement dans l'espace RGB.

Au contraire, la répartition des pixels de teinte chair dans l'espace couleur RG-normalisé, figure 5.2 a l'avantage d'être plus localisée, et donc modélisable plus simplement. Le passage d'une couleur (r, g, b) de l'espace RGB vers l'espace RG-normalisé s'effectue simplement en divisant chaque composante par la somme des trois composantes :

$$r' = \frac{r}{r+g+b} \tag{5.1}$$

$$g' = \frac{g}{r+g+b} \tag{5.2}$$

On perd dans cet espace l'information sur la somme des 3 composantes, qui approxime en réalité l'intensité de la couleur. Les valeurs r' et g' sont les composantes de chrominance de la couleur, et suffisent à représenter correctement les pixels de teinte chair. En effet, la peau possède une chrominance très localisée, mais peut prendre des valeurs de luminance variées, selon l'origine ethnique de chaque individu et l'éclairage de la scène.

La répartition de la teinte chair dans l'espace RG-normalisé est supposée Gaussienne, ce qui paraît raisonnable à première vue d'après le nuage de points figure 5.2. Ce modèle a l'avantage d'être paramétré par seulement 5 scalaires, qui sont les moments de premier ordre $\boldsymbol{\mu}_P$ et de second ordre Γ_P :

$$\boldsymbol{\mu}_P = \begin{bmatrix} \mu_r \\ \mu_g \end{bmatrix} \tag{5.3}$$

$$\Gamma_P = \begin{bmatrix} \gamma_{rr} & \gamma_{rg} \\ \gamma_{rg} & \gamma_{gg} \end{bmatrix} \tag{5.4}$$

La densité de probabilité f_P de la teinte chair s'exprime alors de manière explicite pour toute couleur $\boldsymbol{c}$ de l'espace RG-normalisé comme :

$$f_P(\boldsymbol{c}) = \frac{1}{2\pi|\Gamma_P|^{1/2}} \exp\left(-\frac{1}{2}(\boldsymbol{c}-\boldsymbol{\mu}_P)^T\Gamma_P^{-1}(\boldsymbol{c}-\boldsymbol{\mu}_P)\right) \tag{5.5}$$

5.1.1.2 Discussion sur le choix du modèle

Nous choisissons de représenter la teinte chair par un modèle Gaussien dans l'espace couleur RG-normalisé. Il est clair qu'aucun phénomène physique ne permet de justifier que la répartition des couleurs de la peau suive une loi normale. En réalité, le modèle gaussien est davantage choisi pour des raisons de facilité d'apprentissage des paramètres et d'utilisation, que pour modéliser les données au plus proche. Les visages des séquences vidéo que nous devons traiter présentent des couleurs fortement biaisées par des facteurs liés au système d'acquisition et à l'environnement. Ces facteurs sont les contrôles automatiques de gain et de balance des couleurs, les variations éclairages naturel et artificiels ou encore la compression JPEG. Ils sont tels que la modélisation précise de la teinte chair est un problème qui n'est pas réellement important, car de nombreuses fausses détections ou non détections surviendront pendant l'étape de détection quelque soit la complexité ou la précision du modèle. En effet, les variations de couleur de peau visibles dans les séquences vidéo réelles sont majoritairement due aux conditions d'acquisition, beaucoup plus qu'aux différences ethniques.

5.1.1.3 Apprentissage

L'apprentissage des paramètres $\boldsymbol{\mu}_P$ et Γ_P du modèle est réalisé sur une base d'apprentissage contenant un grand nombre d'exemples de pixels de teinte chair. Ces pixels ont été extraits d'images issues de la base FERET [PWHR98].

Cette base de données contient les photos de 1208 sujets, prises avec des conditions d'éclairage et des orientations du visage variées. La position des yeux, du nez et de la bouche sont annotées pour chaque photo, ce qui nous permet d'extraire automatiquement de l'image une zone contenant principalement des pixels de peau. On réalise pour cela un découpage automatique d'une zone rectangulaire de l'image comprise entre les yeux et la bouche, comme illustré sur la figure 5.3. Il en résulte un ensemble d'environ 56 millions de pixels de peau, qui constituent notre base d'apprentissage pour les paramètres du modèle.

L'estimation des paramètres est simple avec le modèle gaussien choisi. Il suffit en effet de calculer le vecteur moyen et la matrice de covariance de la base d'apprentissage.

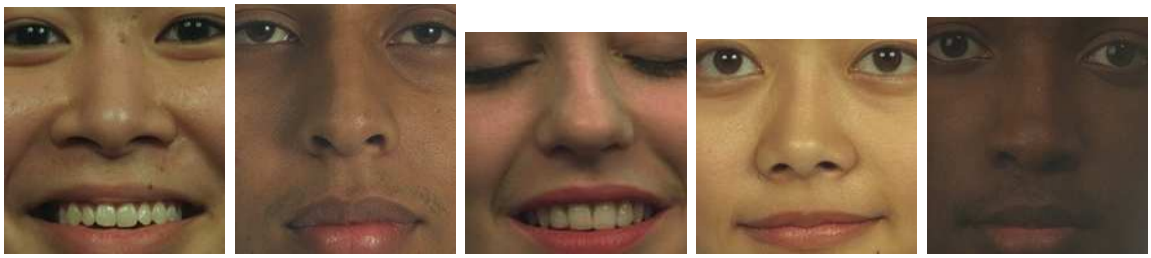


FIG. 5.3 – Échantillons de peau issus de la base d'images FERET



(a) (b)

FIG. 5.4 – Détection de peau. (a) : Image originale (b) : Carte de probabilité de teinte chair

5.1.2 Détection

La détection des pixels de teinte chair dans une image I donnée consiste à déterminer pour chaque pixel de l'image la probabilité que ce pixel puisse être issu de la distribution de teinte chair précédemment modélisée. On construit alors la carte de probabilités de peau comme proportionnelle à $f_P(I)$, de façon à ce que la valeur en chaque point de la carte de probabilité soit proportionnelle à la densité de peau pour la couleur du pixel considéré.

La figure 5.4 présente un exemple de détection de peau obtenue sur une séquence réelle de véhicule. Les valeurs plus claires correspondent aux pixels dont la probabilité de peau est plus forte. On peut constater que les zones de teinte chair comprennent non seulement les visages des passagers, mais aussi leurs mains et bras, ainsi que certains éléments de l'arrière-plan qui ne sont pas de la peau. Le système d'acquisition crée aussi des couleurs similaires à la peau, notamment sur les contours aux forts contrastes.

5.1.3 Mesure de la performance de détection

Afin de caractériser la performance de détection de la teinte chair en fonction des différents paramètres influant sur la couleur de l'image, nous avons mené une expérience sous conditions contrôlées. Les facteurs testés sont la balance des couleurs et le taux de compression JPEG.

Une série de photos de mains a été acquise, avec des arrière-plans et des éclairages de couleurs variées (figure 5.5). Un soin particulier a été porté au placement de la main pour chaque photo, afin qu'une vérité terrain des pixels de peau puisse être obtenue facilement. Les arrières plans utilisés ont été volontairement choisis avec des couleurs très vives, et les photos résultantes montrent clairement le travail du contrôle automatique des balance des couleurs. Les couleurs étranges résultantes pour les pixels de peau mettent bien entendu en difficulté la détection de la peau par notre modèle de couleur. Les taux de bonne détection et de fausse détection obtenus par notre modèle de peau, par rapport à la vérité terrain, sont précisés dans la figure 5.5. On note que la couleur de l'éclairage (qui diffère sur les quatre premières photos) a une influence importante sur la performance de la détection.

L'autre expérience réalisée cherche à caractériser l'influence du taux de compression JPEG sur la détection de peau. L'information de chrominance dans l'image est fortement dégradée lors d'une compression JPEG, car c'est une information qui a moins d'importance que la luminance dans l'impression de qualité donnée par l'image résultante.

La figure 5.6 illustrent les 20 images utilisées pendant cette expérience. Pour que l'expérience soit plus proche des conditions réelles, on incruste en arrière plan des images de mains une photo de milieu urbain. Les taux de bonne détection et fausse détection correspondant sont présentés dans la figure 5.7. La détection de la teinte chair semble donc peu influencée par le taux de compression JPEG.

5.2 Détection des parties vitrées

Un élément important dans l'analyse de la scène concerne la détection des parties vitrées du véhicule. L'extraction de cette information est en effet une première étape vers des algorithmes de plus haut-niveau. En particulier, à l'image les parties vitrées ont des comportements très différents des autres parties du véhicule, et requièrent parfois un traitement particulier pour qu'une certaine tâche puisse être accomplie sur l'ensemble de l'image. Par exemple la détection de pixels en mouvement est une tâche pour laquelle certains algorithmes ne fonctionnent que sur les parties de la vue correspondant à l'intérieur du véhicule.

L'autre intérêt d'extraire les parties vitrées est qu'il s'agit d'une première approche pour la détection de la porte et de son ouverture. Cette application sera détaillée dans la suite.

Photo	Bonne détections	Fausses détections
	97.4%	1.61%
	87.3%	6.2%
	91.9%	9.0%
	85.0%	6.8%
	0%	100%
	10.2%	31.7%
	0%	100%
	41.2%	4.2%
	0%	100%
	3.6%	53.6%

Fig. 5.5 – Photos d'une main avec des arrière-plans et éclairages de couleurs variées. Taux de bonnes détections/fausses détections

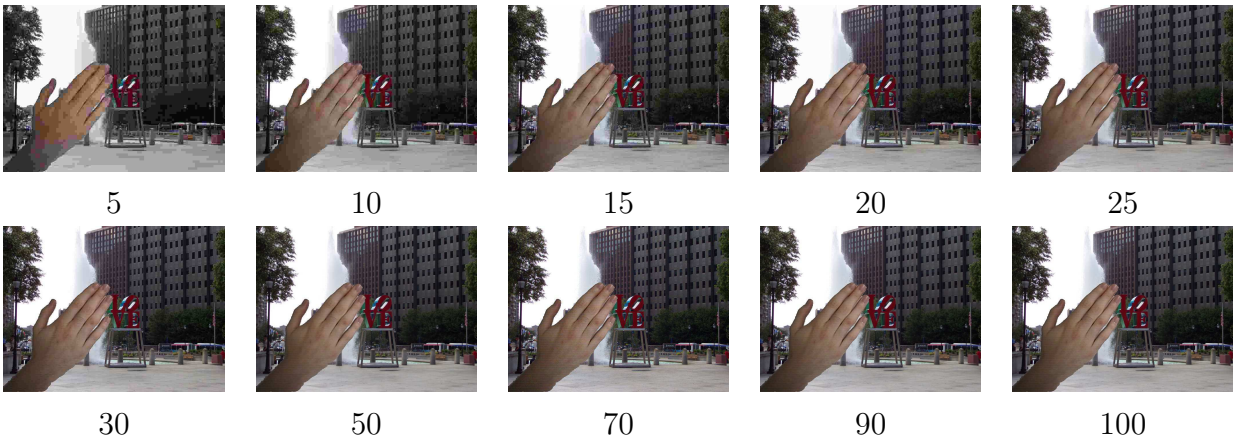


FIG. 5.6 – Photos d'une main compressées en JPEG avec différents facteurs de qualité Q

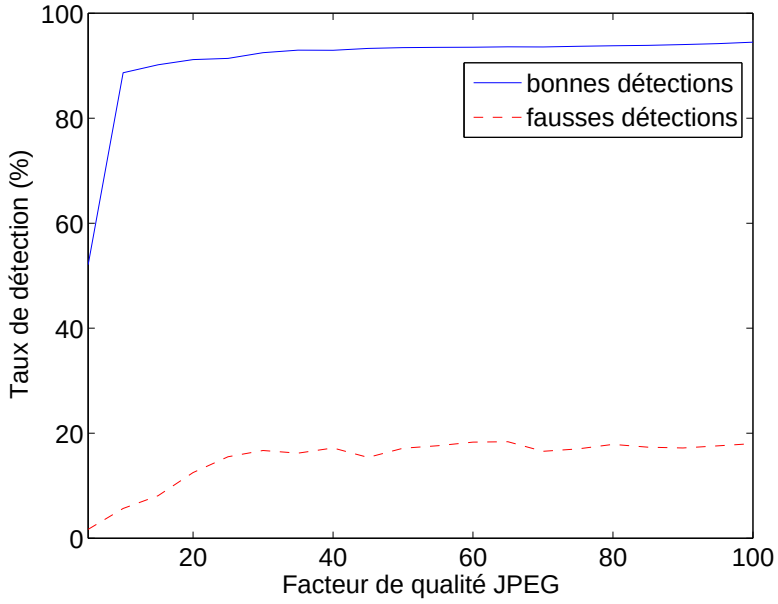


FIG. 5.7 – Performance de la détection de teinte chair en fonction du taux de compression JPEG

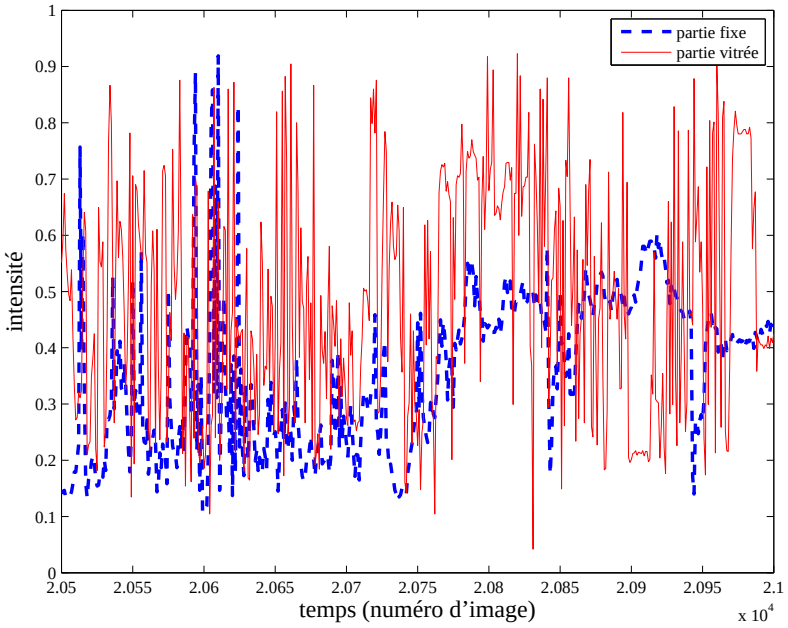


FIG. 5.8 – Profils temporels de pixels de la partie fixe et la partie vitrée

5.2.1 Quantification de la variation de chaque pixel

Comme le véhicule est mobile la plupart du temps, la variation de la valeur de chaque point en fonction du temps semble être une information importante pour discerner les parties vitrées du reste du véhicule. Bien que la scène comporte des variations en tout point, à cause des diverses perturbations telles que les ombres portées ou le mouvement des passagers, on peut supposer que le mouvement important visible dans les parties vitrées entraîne une variation plus rapide et plus fréquente de la valeur des pixels. Cette constatation est vraie globalement sur une séquence vidéo longue, de l'ordre d'une journée, et peut être fausse sur des courtes durées, par exemple pendant un arrêt. L'extraction des parties vitrées s'effectue donc sur une longue durée, en intégrant l'information apportée par chaque image.

5.2.1.1 Comparaison de la partie vitrée et de la partie fixe

La figure 5.8 présente les profils temporels de deux pixels appartenant chacun à une classe différente (partie vitrée et partie fixe). Seule l'intensité des pixels est représentée ici.

Une première remarque que l'on peut tirer de ces profils, est que les pixels des deux classes présentent des variations en intensité, sur une plage de valeur équivalente. Ceci est dû aux forts changements que peuvent prendre les pixels de la partie fixe, entre les instants où le véhicule est plongé dans l'ombre et les instants où il se trouve en plein soleil. La variance du signal temporel n'est donc pas une information assez discriminante.

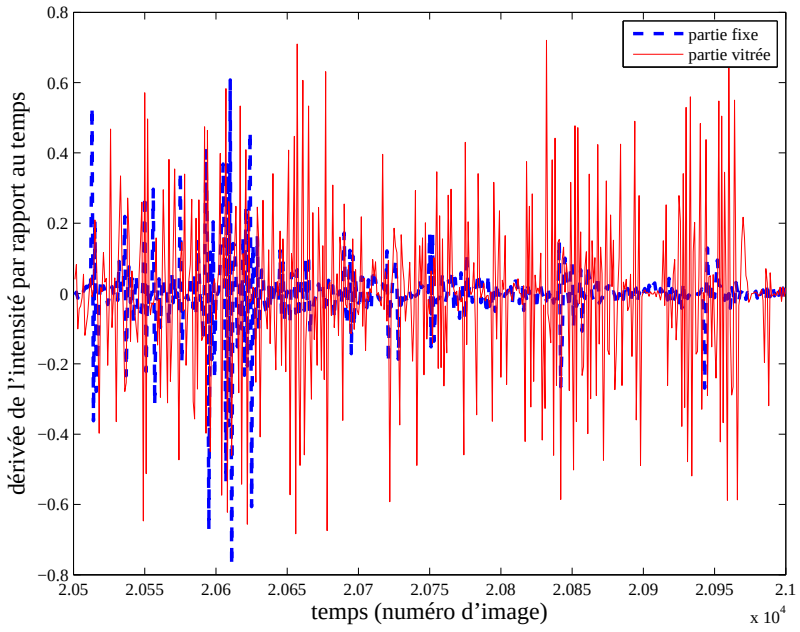


FIG. 5.9 – Dérivée de l'intensité des profils temporels de pixels de la partie fixe et la partie vitrée

On peut toutefois noter les variations plus rapides et plus fréquentes du profil temporel du pixel de la partie vitrée. Les contenus fréquentiels de chacun des signaux semblent être assez différents, avec des hautes fréquences beaucoup plus présentes pour la partie vitrée. Ceci nous incite à considérer le signal dérivé de l'intensité, afin de filtrer les basses fréquences qui ne semble pas apporter d'information pertinente (figure 5.9).

5.2.1.2 Mesure du contenu haute-fréquence

Une façon simple de mesurer le contenu haute fréquence du profil temporel d'un pixel $\boldsymbol{k}$ est de considérer la puissance moyenne du signal dérivé :

$$P(\boldsymbol{k}) = \frac{1}{t_2 - t_1} \int_{t_1}^{t_2} |y_{t+1}(\boldsymbol{k}) - y_t(\boldsymbol{k})|^2 dt \tag{5.6}$$

avec t_1 et t_2 les temps de début et fin de la séquence, et $y_t(k)$ l'intensité du pixel $\boldsymbol{k}$ au temps t.

Le résultat est illustré dans la figure 5.10, représentant la puissance moyenne pour chaque pixel sur toute la durée de la séquence. Les puissances moyennes sont normalisées entre 0 et 1, et composent ainsi la carte de probabilité d'appartenance à une vitre. Cela implique qu'on fasse l'hypothèse de la présence d'au moins une région vitrée dans la vue. La figure 5.10 montre qualitativement que la partie vitrée est extraite correctement. D'autres résultats de détection, pour des caméras différentes sont présentés dans la figure 5.11.



FIG. 5.10 – Puissance moyenne de la dérivée du profil temporel de chaque pixel

Aucune décision de type seuillage n'est réalisée afin de conserver l'incertitude sur la classe de certains pixels.

5.2.2 Application à la localisation de la porte du véhicule

Pour certaines applications de notre contexte, il peut être intéressant de localiser précisément la porte du véhicule à l'image. Il s'agit par exemple d'une première étape vers un algorithme permettant de détecter une ouverture de porte, et d'ainsi séparer les séquences d'arrêt du véhicule des autres séquences vidéo. Pour une application de suivi de personnes, cela peut aussi servir à guider le détecteur de personne vers la zone où les passagers sont le plus susceptible d'apparaître à l'image. Un algorithme de localisation de la porte du véhicule est donc introduit ici, basé sur la détection des parties vitrées qui vient d'être présentée.

5.2.2.1 Définition du modèle de porte

La première étape consiste à détecter quelle zone de la partie vitrée correspond à la porte. Comme le système doit fonctionner dans un grand nombre de véhicules différents avec des positions de caméra relativement différentes, on peut faire très peu d'hypothèses sur la position de la porte. La porte est toutefois définie par les hypothèses suivantes, qui restent suffisamment générales :

1. La porte projetée dans l'image est en forme de quadrilatère.
2. La porte contient une zone vitrée sur toute sa hauteur.
3. L'extérieur de la porte n'est pas vitré
4. La hauteur de cette zone vitrée est supérieure à celle des fenêtres qui peuvent aussi être présentes.

Les portes sont souvent composées de deux battants. Nous nous limitons à la détection d'un seul battant, ce qui est suffisant pour la détection de l'ouverture.

La zone vitrée de la porte est modélisée par un quadrilatère, paramétrisé par les coordonnées des 4 sommets. Nous nous plaçons dans un cadre Bayesien, dans lequel le résultat de la détection de la partie vitrée est l'observation $\boldsymbol{z}$, tandis que $\boldsymbol{x}$ désigne une hypothèse de porte, issue d'une variable aléatoire dont la densité de probabilité doit être définie. La loi de Bayes nous indique que

$$p(\boldsymbol{x}/\boldsymbol{z}) \propto p(\boldsymbol{z}/\boldsymbol{x})p(\boldsymbol{x}) \tag{5.7}$$

5.2.2.2 Estimation des paramètres par simulation Monte-Carlo

La nature multimodale de la densité *a posteriori*, due à la présence importante d'éléments de la scène ressemblant à une porte, ne permet pas d'utiliser une méthode d'estimation simple telle qu'une descente de gradient. Afin d'approcher la densité de probabilité *a posteriori* $p(\boldsymbol{x}/\boldsymbol{z})$, et de déterminer par la suite les paramètres les plus probables pour la porte, une simulation Monte-Carlo est effectuée sur la variable aléatoire de la densité *a priori* $p(\boldsymbol{x})$.

Probabilité *a priori* Cette densité *a priori* est définie explicitement, suivant des critères de taille et de position déterminés empiriquement. Un ensemble de lois uniformes est alors défini :

1. Le centre du quadrilatère suit une loi uniforme définie sur les positions de l'image où le centre de la porte est susceptible de se situer.
2. La hauteur de la porte suit une loi uniforme.
3. Le quotient de la largeur de la porte sur la hauteur de la porte suit une loi uniforme sur un intervalle adéquat, par exemple [0.20.8].
4. La position de chacun des 4 sommets du quadrilatère est une loi uniforme dont le domaine de définition est fonction de chacun des 4 sommets du rectangle de même centre et de mêmes dimensions.

Le tirage aléatoire d'un quadrilatère $\boldsymbol{x}$ commence donc par un tirage aléatoire du centre, de la hauteur et de la largeur. Un rectangle peut alors être associé à ces paramètres. Notons $\boldsymbol{r}_1, \ldots, \boldsymbol{r}_4$ ses 4 sommets. Chaque sommet $\boldsymbol{q}_k, k = 1 \ldots 4$ du quadrilatère $\boldsymbol{x}$ est alors tiré à partir d'une loi uniforme dont le domaine de définition est centré sur le point $\boldsymbol{r}_k$. Les domaines de définition exacts de chaque loi uniforme ne sont pas mentionnés ici, ils sont déterminés empiriquement, de telle façon à ne pas générer de quadrilatère trop improbables pour une porte.

Probabilité d'observation Un nombre N_h d'hypothèses $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{N_h}$ est généré à partir de la variable aléatoire définie précédemment. La probabilité d'observation $p(\boldsymbol{z}/\boldsymbol{x})$ de chaque hypothèse est alors calculée, à partir du résultat de détection des parties vitrées. $p(\boldsymbol{z}/\boldsymbol{x})$ est définie explicitement, suivant des critères empiriques, choisis de manière à ce que le maximum soit obtenu pour les paramètres réels de la porte.

La probabilité d'observation proposée prend en compte deux critères utilisée conjointement :

1. L'intérieur du quadrilatère est une zone vitrée
2. L'extérieur du quadrilatère n'est pas une zone vitrée

Le second critère permet de s'assurer que $p(\boldsymbol{z}/\boldsymbol{x})$ n'atteigne pas un maximum pour un quadrilatère situé à l'intérieur d'une zone vitrée.

Pour chaque hypothèse $\boldsymbol{x}_i$, un quadrilatère $\bar{\boldsymbol{x}}_i$ est défini autour du quadrilatère $\boldsymbol{x}_i$, de telle façon que $\bar{\boldsymbol{x}}_i$ contienne le quadrilatère $\boldsymbol{x}_i$ ainsi que la zone autour de $\boldsymbol{x}_i$ qui ne doit pas être vitrée.

Notons $q(\boldsymbol{x})$ l'ensemble des points de l'espace à l'intérieur d'un quadrilatère paramétrisé par $\boldsymbol{x}$, et considérons maintenant $\mathcal{A}(\boldsymbol{x})$ la valeur moyenne des puissances moyennes $P(k)$ (équation 5.6) pour tout pixel k à l'intérieur de $\boldsymbol{x}$:

$$\mathcal{A}(\boldsymbol{x}) = \frac{1}{Card(q(\boldsymbol{x}))} \sum_{k \in q(\boldsymbol{x})} P(k) \tag{5.8}$$

La probabilité d'observation d'une hypothèse de porte $\boldsymbol{x}_i$ est définie comme suit :

$$p(\boldsymbol{z}/\boldsymbol{x}_i) \propto \mathcal{A}(\bar{\boldsymbol{x}}_i) - \mathcal{A}(\boldsymbol{x}_i) \tag{5.9}$$

Maximum *a posteriori* Par une application directe du théorème de Bayes, équation 5.7, l'ensemble des hypothèses $\boldsymbol{x}_i$, générées par la variable aléatoire de la densité *a priori*, et pondérées par leur probabilité d'observation $p(\boldsymbol{z}/\boldsymbol{x}_i)$, forme une approximation de la densité de probabilité *a posteriori* $p(\boldsymbol{x}/\boldsymbol{z})$. L'hypothèse $\hat{\boldsymbol{x}}$ dont la probabilité d'observation est maximum est l'estimateur du maximum *a posteriori* (MAP).

La figure 5.11 illustre le résultat de la localisation de la porte, pour trois caméras différentes. Le nombre d'échantillons utilisés dans nos simulations pour la méthode de Monte-Carlo est de l'ordre du million, ce qui demande un temps de calcul d'environ 30 secondes. Le temps réel n'est bien entendu pas requis, vu que la localisation ne doit être effectuée qu'une seule fois, pour une vue de caméra donnée.

5.3 Détection des pixels d'avant-plan par des méthodes classiques

Le mouvement est une information qui est classiquement utilisée pour la détection d'objets vidéo d'intérêt. Dans un contexte où la caméra est fixe par rapport à la scène filmée, il est généralement considéré que les objets d'intérêt sont mobiles et une détection de mouvement permet de les extraire des données vidéos. Une des applications principales que nous souhaitons réaliser pour notre contexte de vidéosurveillance embarquée est la détection des personnes dans le véhicule. Dans ce contexte, la caméra est effectivement fixe par rapport à son environnement, mais il faut bien sûr tenir compte des nombreuses perturbations de la



FIG. 5.11 – Détections de portes (quadrilatères blancs) à partir des cartes de probabilité d'appartenance à une vitre

scène. En particulier, l'arrière-plan est divisé en deux parties aux comportements temporels très différents : l'intérieur du véhicule et le paysage extérieur. Les méthodes classiques de détection de mouvement ne peuvent donc pas s'appliquer directement pour l'extraction des personnes dans la vidéo. Néanmoins, un bref récapitulatif de ces méthodes ainsi qu'une étude de leur applicabilité à la détection de personnes va aider à définir une méthode pour la détection des personnes basée sur le mouvement.

Par méthode de détection basée sur le mouvement, on entend au sens large les méthodes permettant de détecter les pixels de la vidéo qui nous intéressent par leur propriétés de mouvement. Cette précision permet d'inclure dans les méthodes de détection de mouvement celles qui sont basées sur l'estimation de l'arrière-plan de la scène. Ces méthodes sont capables, dans des conditions adaptées, de détecter les objets d'intérêt d'avant-plan même lorsqu'ils sont statiques. Les méthodes de détection de mouvement sont alors regroupées en deux familles :

- Les méthodes basées sur la détection du mouvement inter-image supposent que les objets d'intérêt que l'on souhaite extraire sont perpétuellement mobiles, et tentent de les extraire en analysant les variations entre images consécutives.
- Les méthodes basées sur une modélisation de l'arrière-plan supposent que les objets d'intérêt sont ceux dont les pixels présentent un mouvement anormal par rapport au fond. Un modèle d'arrière-plan est obtenu, et comparé à chaque image de la séquence pour déterminer quels pixels appartiennent à l'avant-plan et à l'arrière-plan.

5.3.1 Détection du mouvement inter-image

La façon la plus triviale de détecter le mouvement entre deux images consécutives consiste à réaliser la soustraction de ces deux images. Il s'agit de la méthode prédictive (comme définie dans le chapitre d'introduction à cette partie) la plus simple, où l'arrière-plan est supposé constant entre deux images, à des variations près qui sont supposées de magnitude inférieure à un seuil donné. L'arrière-plan est supposé statique, ce qui résulte en des valeurs quasi-nulles lors de la soustraction. Les objets d'intérêt sont supposés mobiles et texturés de façon à ce que la soustraction résulte en des valeurs relativement fortes. Comme les images traitées sont des images couleurs, on remplace la soustraction par une mesure de distance entre couleurs, mais le principe et le type de résultat restent les mêmes. La distance considérée est simplement la distance euclidienne dans l'espace couleur RGB, initialement utilisé dans nos images.

Comme illustré sur la figure 5.12, la soustraction inter-image ne convient pas pour détecter les objets d'intérêt (c'est-à dire ici les personnes) dans nos séquence réelles. Les objets ne sont pas toujours mobiles, et apparaissent mal dans le résultat. D'autre part, d'autres objets de la scène présentent du mouvement et apparaissent dans la soustraction, comme par exemple les parties vitrées.

(a)



(b)



(c)

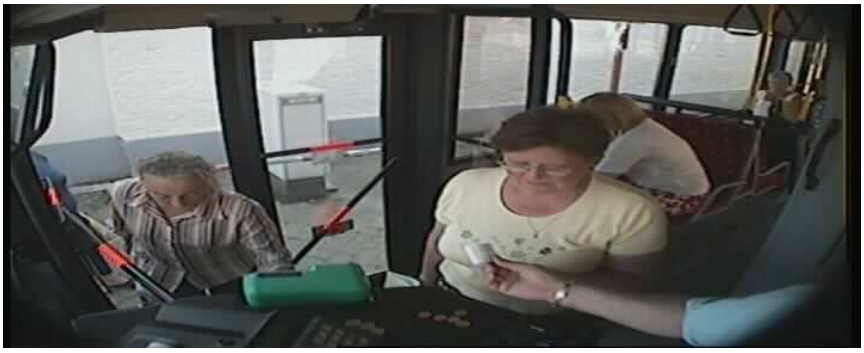


FIG. 5.12 – Soustraction entre images consécutives. (a) et (b) : Images originales. (c) : Résultat de la soustraction



FIG. 5.13 – Moyenne temporelle des images d'une séquence vidéo sur deux jours

(a)



(b)



FIG. 5.14 – (a) : image originale. (b) : soustraction entre image moyenne et image originale

5.3.2 Modélisation du fond

5.3.2.1 Image moyenne

La modélisation de l'arrière-plan est une approche séduisante pour la détection des objets d'intérêt, dont l'étude a été motivée par le constat que les méthodes les plus basiques procurent déjà des résultats intéressants et prometteurs sur nos séquences vidéo. Un simple moyennage temporel des images de la séquence produit une estimation précise de l'intérieur du véhicule. La figure 5.13 illustre la moyenne temporelle des images d'une séquence prise sur deux jours d'exploitation. Seuls les éléments statiques de la scène transparaissent dans le résultat. En particulier, les passagers du véhicule ne sont pas visibles car leur présence en tant qu'objets statiques pendant le trajet est négligeable par rapport au temps d'intégration de 2 jours. Le paysage extérieur, comportant de fortes variations tout au long de la séquence, n'est pas non plus discernable, la moyenne temporelle des pixels des parties vitrées ayant convergé vers le gris.

La soustraction d'une image de la séquence et de l'image moyenne fait apparaître tous les éléments de la scène qui ne sont pas statiques (figure 5.14). Il s'agit de la méthode de soustraction de fond de base, proposée par [LY87], mais avec une extraction de l'arrière-plan par la moyenne temporelle et un espace couleur RGB au lieu de seulement l'intensité. Comme escompté, cette détection de mouvement n'est pas satisfaisante car elle détecte aussi bien les passagers que les ombres portées et le paysage extérieur. De plus, les variations de luminosité globale de la scène, provoquées par l'éclairage intérieur au véhicule ou

simplement l'instant de la journée, ne sont pas prise en compte par ce modèle de fond. Ces résultats sont donc difficilement exploitables.

5.3.2.2 Modélisation Gaussienne des variations

On se dirige donc vers des modèles plus élaborés, prenant en compte les différentes variations possibles de l'arrière-plan. Un modèle simple consiste à supposer un comportement gaussien pour chaque position spatiale, comme proposé par [WADP97]. Il est supposé qu'un pixel de la séquence prend des valeurs au cours du temps qui sont issues d'une distribution gaussienne. Les paramètres de chaque gaussienne sont estimés sur une séquence vidéo d'apprentissage prise dans le même véhicule, en situation réelle, afin que les variations des pixels soient les plus proches possibles des variations attendues lors de l'étape de détection.

En prenant en compte les variations de chaque pixel au cours du temps, on espère s'affranchir des problèmes liés aux parties vitrées du véhicule, pour lesquelles la moyenne des valeurs prises par chaque pixel n'a pas de sens. Les pixels des parties vitrées ont des variations qui sont très fortes, de telle façon qu'il n'est pas envisagé de pouvoir différencier les passagers présents devant les vitres du véhicule, du paysage extérieur. En contrepartie, une modélisation correcte de l'intérieur du véhicule permettrait de discriminer les pixels du véhicule des pixels des passagers dans le véhicule, qui seront considérés comme en dehors du modèle.

La figure 5.15 illustre les histogrammes des couleurs visibles au cours de la séquence d'apprentissage, pour trois positions spatiales. Contrairement à nos attentes, l'histogramme du point B, qui est un pixel appartenant à la partie vitrée, montre que la variation de ce point n'est pas très importante, et qu'elle est même moins importante que pour certains autres points de l'intérieur du véhicule. Ce phénomène semble provenir du fait que les parties vitrées apparaissent grises la plupart du temps à cause de la couleur de la route. On fera d'ailleurs ici le parallèle avec l'étude préliminaire de la variation temporelle des pixels, menée pour l'extraction des vitres présentée précédemment. La variation en intensité n'est pas une mesure discriminante entre les régions vitrées et les régions non-vitrées (figure 5.8). Les histogrammes de la figure 5.15 montrent qu'il en est de même pour les composantes de chrominance dans l'espace RG-normalisée.

Le modèle gaussien adopté pour chaque pixel est défini pour l'espace couleur RGB et pour l'espace RG-normalisé. L'estimation de ses paramètres pour une position $\boldsymbol{k}$ consiste à calculer les moyennes de chaque composante ainsi que la matrice de covariance des valeurs prises. Le nombre de paramètres scalaires du modèle est de 5 pour l'espace couleur RG-normalisé, et de 9 pour l'espace couleur RGB. En notant $\boldsymbol{\mu_k}$ le vecteur moyen et $\Gamma_{\boldsymbol{k}}$ la matrice de covariance, les paramètres estimés sont :

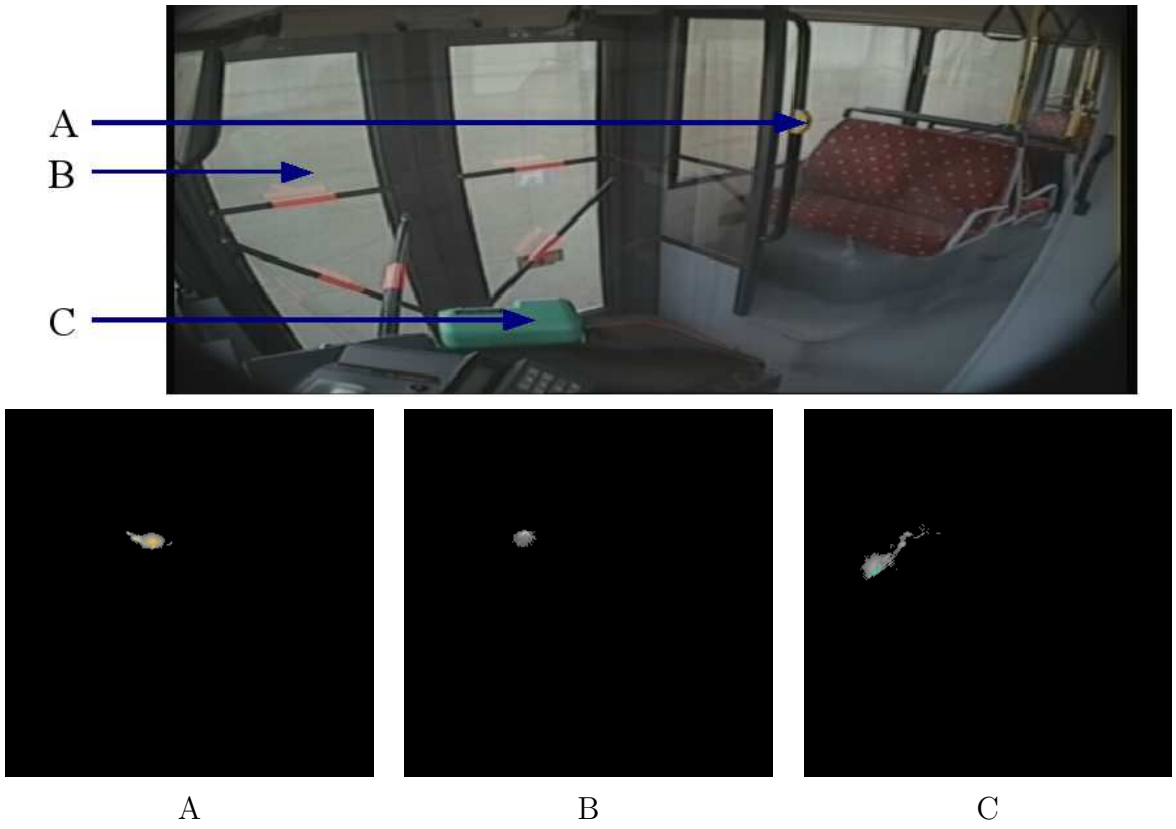


FIG. 5.15 – Histogrammes de couleurs en RG-normalisé des valeurs prises par différents pixels au cours du temps

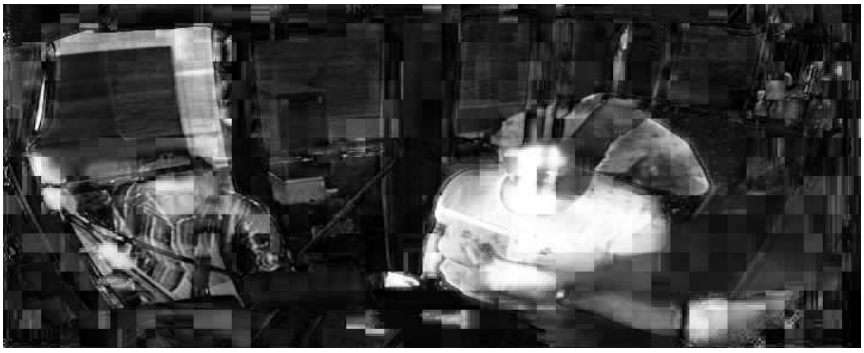


FIG. 5.16 – Détection des pixels d'intérêt avec le modèle gaussien d'arrière-plan

$$\begin{aligned} \boldsymbol{\mu_k} &= \frac{1}{T}\sum_{t=1}^{T} I_t(\boldsymbol{k}) \\ \Gamma_{\boldsymbol{k}} &= \frac{1}{T}\sum_{t=1}^{T} I_t(\boldsymbol{k}).I_t(\boldsymbol{k})^T \end{aligned} \tag{5.10}$$

avec I_t l'image au temps t et T le temps total de la séquence d'apprentissage. L'apprentissage a été effectué sur une durée de 2 jours d'enregistrement, sachant que le système de surveillance n'était pas en fonctionnement pendant la nuit.

5.3.2.3 Détection des pixels d'avant-plan

La détection des pixels des objets d'intérêt consiste alors à déterminer pour chaque pixel $\boldsymbol{k}$ de l'image courante I_t sa probabilité $p(\boldsymbol{k} \in B)$ d'appartenance au modèle d'arrière-plan. On définit cette probabilité en fonction de la distance de Mahalanobis de la couleur du pixel considéré à la distribution estimée des couleurs d'arrière-plan :

$$p(\boldsymbol{k} \in B) \propto exp\left(-\frac{1}{2} I_t(\boldsymbol{k}) \Gamma_{\boldsymbol{k}}^{-1} I_t(\boldsymbol{k})^T\right) \tag{5.11}$$

La probabilité pour qu'un pixel $\boldsymbol{k}$ appartienne à un objet d'intérêt est alors simplement définie comme :

$$p(\boldsymbol{k} \in F) = 1 - p(\boldsymbol{k} \in B) \tag{5.12}$$

La figure 5.16 présente la carte de probabilité que l'on obtient avec ce modèle gaussien d'arrière-plan, pour l'image originale de la figure 5.14(a). La détection des pixels de passagers est légèrement améliorée, même si les fausses détections sont toujours là. Notamment, la porte du véhicule, ouverte, est détectée comme en dehors du modèle, celui-ci ayant été appris principalement lorsque la porte était fermée. De plus, la personne sur la gauche n'est pas correctement détectée. La détection obtenue n'est toujours pas de qualité suffisante pour être exploitable.

Ces résultats montrent que les variations de la scène ne sont très difficilement modélisables, et perturbent l'extraction des pixels d'avant-plan. De plus, les objets d'intérêts ont des couleurs qui sont souvent indifférenciables de l'arrière-plan. Ceci nous incite à considérer une autre méthode pour la détection des pixels de passagers. La méthode présentée dans la suite est toujours basée sur une modélisation de l'arrière-plan, mais est spécifique aux situations d'arrêts du véhicule.

5.4 Estimation du fond de la scène à un arrêt

La majeure partie des applications que l'on peut envisager dans notre contexte et qui nécessitent une détection de personnes peuvent se contenter du seul traitement des séquences d'arrêt du véhicule. En se limitant aux séquences d'arrêt, les contraintes changent. En effet, le problème de détection de mouvement dans les parties vitrées disparaît en grande partie, ainsi que le problème des changements locaux brusques d'illumination dus aux ombres portées. En contrepartie, les séquences à traiter sont beaucoup plus courtes qu'auparavant, et le nombre de personnes présentes devant la caméra peut être excessivement élevé. Cela nécessite l'élaboration de méthodes spécifiques à ce type de séquences.

5.4.1 Caractérisation des séquences d'arrêt du véhicule

Une scène typique d'arrêt du véhicule, vue à travers la caméra chauffeur dirigée vers la porte avant, montre l'apparition de passagers, leur montée dans le véhicule, puis leur disparition du champ de la caméra. Ce type de séquences a une durée très courte, d'environ une trentaine de secondes, pendant lesquelles les passagers doivent être détectés. Nous nous intéressons pour l'instant à la seule détection des pixels de la séquence appartenant à des passagers. La détection des passagers à proprement parler, en tant qu'objets, sera une seconde étape. Afin de définir correctement un algorithme capable d'extraire les pixels des passagers, il est important de comprendre quelles sont les spécificités des séquences d'arrêt.

Comme dans le cas général des séquences du véhicule en mouvement, les perturbations de la scène peuvent être classées en variations d'illumination et mouvements d'objets.

5.4.1.1 Variations d'illumination

Les séquences d'arrêt étant généralement courtes, les variations d'illumination dues aux changements jour/nuit sont négligeables. De plus, le véhicule étant à l'arrêt, les ombres portées sont stables et n'introduisent pas de perturbation supplémentaire dans la séquence. Pourtant, le fond de la scène ne peut pas être considéré statique à cause des contrôles automatiques de la caméra, en gain et en balance des couleurs, qui introduisent des variations rapides en luminance et en teinte. Ces variations sont provoquées par l'apparition des passagers devant la caméra, qui modifient le contenu de l'image, auquel s'adaptent directement les contrôles automatiques du système d'acquisition.

5.4.1.2 Mouvements d'objets

La principale source de mouvement dans la scène est évidemment causée par la présence des passagers devant la caméra. La fonction première de la caméra est de permettre la surveillance de l'activité dans le véhicule. Par conséquent, sa position est telle que les passagers apparaissent à l'image en gros plan. La proportion de pixels de la séquence appartenant à des personnes est très importante, de l'ordre de 50%.

D'autre part, bien que le véhicule est à l'arrêt, la caméra n'est pas complètement statique par rapport à l'environnement extérieur. En effet, la montée des personnes introduit un léger déplacement vertical dû aux suspensions du véhicule, qui peut se traduire à l'image par une translation de quelques pixels (de l'ordre de 5 pixels). Ce déplacement n'est pas un déplacement global sur l'image ; il ne concerne que les parties vitrées, dont la proportion à l'image dépend de chaque véhicule, pouvant aller jusqu'à plus de la moitié de la surface de l'image.

Une autre source de mouvement concernant les séquences d'arrêt est créée par les éléments dynamiques de l'environnement extérieur, tels que du feuillage en présence de vent. Ces éléments génèrent des variations assez faibles, mais suffisantes pour perturber le fonctionnement d'algorithmes classiques, comme nous le verrons lors de la comparaison de la solution proposée avec les mélanges de Gaussiennes.

L'ensemble de ces perturbations dans la vidéo font que l'on peut se considérer en présence d'une scène dynamique, pour laquelle il faut envisager une méthode spécifique de détection des pixels appartenant aux personnes. La méthode proposée dans la suite modélise certaines perturbations et estime leurs paramètres, tandis qu'elle est simplement robuste à d'autres perturbations en utilisant des caractéristiques de l'image qui leur sont invariantes.

5.4.2 Présentation générale de la méthode proposée d'estimation du fond à un arrêt

Afin de détecter les pixels de la vidéo appartenant à des personnes, nous passons par l'estimation du fond de la scène, qui malgré les variations importantes présentées ci-dessus, apparaît comme un choix plus judicieux que de se baser uniquement sur la détection de mouvement des passagers. Effectivement, les passagers peuvent souvent se retrouver immobiles pendant un temps non négligeable de la séquence, et leur détection nécessite alors d'avoir une information sur le fond de la scène.

5.4.2.1 Formalisation du problème

Le problème, classique en traitement vidéo, de l'estimation de l'arrière-plan dans une séquence, consiste à extraire automatiquement à chaque image une approximation de la "scène vide", c'est à dire sans les objets d'avant-plan. La séparation entre arrière-plan et avant-plan est généralement dirigée par l'aspect dynamique des éléments de la scène. Dans un cas simple, l'arrière-plan se différencie de l'avant-plan par son absence de variation,

facilement identifiable par rapport à des objets d'avant-plan mobiles. Mathématiquement, on peut exprimer ce problème d'estimation en modélisant la séquence vidéo comme suit :

$$\boldsymbol{y}_t(k) = (1 - i_t(k)).\boldsymbol{bg}_t(k) + i_t(k).\boldsymbol{fg}_t(k) \tag{5.13}$$

au temps t et au pixel k. L'image observée $\boldsymbol{y}_t$ est composée d'une couche d'arrière-plan $\boldsymbol{bg}_t$ et d'une couche d'avant-plan $\boldsymbol{fg}_t$, contrôlées par le processus i_t, tel que $i_t(k) = 1$ lorsqu'un objet d'avant plan est présent en k, et 0 sinon. L'estimation de l'arrière-plan est l'estimation de $\boldsymbol{bg}_t(k)$ pour tout temps t et pixel k. Le processus i_t est généralement inconnu, et la résolution de ce problème implique donc d'émettre des hypothèses sur le fond de la scène ou sur l'avant-plan afin de pouvoir les différencier.

5.4.2.2 Modélisation de l'arrière-plan

Notons $\widehat{\boldsymbol{bg}_{t-1}}$ l'estimation de l'arrière-plan au temps $t-1$. Nous faisons l'hypothèse d'un arrière-plan fixe, sur lequel interviennent des variations en illumination et en mouvement. L'arrière-plan au temps t peut s'écrire en fonction de $\boldsymbol{bg}_{t-1}$ comme :

$$\forall k, \boldsymbol{bg}_t(k) = f_t(\boldsymbol{bg}_{t-1}, k) \tag{5.14}$$

f_t décrivant la transformation globale de l'arrière-plan entre les temps $t-1$ et t. Ce formalisme est certes très général, mais il met en valeur deux points importants :

- La connaissance d'une estimée $\widehat{\boldsymbol{bg}_{t-1}}$ de $\boldsymbol{bg}_{t-1}$ ainsi que de l'image observée $\boldsymbol{y}_t$ permet d'estimer les paramètres de la transformation f_t, à condition d'être robuste aux pixels de $\boldsymbol{y}_t$ appartenant à l'avant-plan.
- La valeur de $\boldsymbol{bg}_t(k)$ ne dépend pas uniquement de $\boldsymbol{bg}_{t-1}(k)$, mais de $\boldsymbol{bg}_{t-1}$, ce qui permet de prendre en compte les mouvement d'éléments de l'arrière-plan.

5.4.3 Transformation linéaire des pixels d'arrière plan

Considérons dans un premier temps une transformation f_t qui modélise uniquement la variation d'illumination de l'arrière-plan entre les temps $t-1$ et t, provoquée par les contrôles automatiques de gain et de balance des couleurs de la caméra. Les mouvements de l'arrière-plan ne sont pas pris en compte pour l'instant. Les contrôles automatiques de la caméra modifient les valeurs des pixels de l'image globalement, de telle façon qu'en tout pixel k, $\boldsymbol{bg}_t(k)$ ne dépend que de $\boldsymbol{bg}_{t-1}(k)$ et des paramètres de la transformation. De plus nous faisons l'hypothèse d'une transformation f_t linéaire. Ce premier modèle s'écrit matriciellement :

$$\forall k, \boldsymbol{bg}_t(k) = A_t.\boldsymbol{bg}_{t-1}(k) + \boldsymbol{e}_t(k) \tag{5.15}$$

$$\forall k, \begin{bmatrix} r_t^{bg}(k) \\ g_t^{bg}(k) \\ b_t^{bg}(k) \\ 1 \end{bmatrix} = A_t . \begin{bmatrix} r_{t-1}^{bg}(k) \\ g_{t-1}^{bg}(k) \\ b_{t-1}^{bg}(k) \\ 1 \end{bmatrix} + \boldsymbol{e}_t(k) \tag{5.16}$$

avec A_t une matrice 4×4 contenant les paramètres de la transformation linéaire. La constante 1 est ajoutée en dernier élément de chaque vecteur de composantes afin que A_t puisse modéliser l'addition d'une constante globalement sur tous les pixels de l'arrière-plan. $\boldsymbol{e}_t(k)$ est l'erreur résiduelle pour le pixel k, par rapport à la prédiction par A_t. Ce modèle de transformation linéaire permet de prendre en compte des changements globaux de luminance ainsi que des changements de chrominance.

5.4.3.1 Estimation des paramètres de la transformation linéaire

Afin d'estimer les paramètres de la transformation linéaire, considérons qu'à l'instant t, nous possédons une estimation $\widehat{\boldsymbol{bg}_{t-1}}$ de l'arrière-plan au temps $t-1$, ainsi qu'une observation $\boldsymbol{y}_t$ de l'image courante, contenant des pixels de l'arrière-plan courant et des objets d'avant-plan.

En récrivant l'équation 5.15 à partir des données, on obtient l'équation suivante :

$$\forall k, \boldsymbol{y}_t(k) = A_t . \widehat{\boldsymbol{bg}_{t-1}}(k) + \boldsymbol{e}_t(k) \tag{5.17}$$

Le terme résiduel $\boldsymbol{e}_t(k)$ contient alors l'erreur d'estimation $\boldsymbol{\delta}_t(k)$ due aux non-linéarités des variations de l'arrière-plan, ainsi que l'erreur, généralement beaucoup plus importante, due aux objets d'avant-plan :

$$\boldsymbol{e}_t(k) = \boldsymbol{\delta}_t(k) + i_t(k).(\boldsymbol{fg}_t(k) - \boldsymbol{bg}_t(k)) \tag{5.18}$$

L'estimation de la transformation linéaire A_t nécessite donc d'être robuste aux objets d'avant-plan, dont les pixels sont considérés comme des *outliers* au modèle. L'utilisation d'une méthode itérative de moindres carrés pondérés [HW77] permet d'estimer correctement A_t. Le principe de la méthode est rappelé dans la suite.

Les pixels d'arrière-plan estimés à l'instant $t-1$ sont regroupés dans une matrice X. De même, les pixels de l'image observée à l'instant t sont regroupés dans la matrice Y. Ces matrices sont organisées comme suit :

$$X = \begin{bmatrix} r_{t-1}^{\widehat{\boldsymbol{bg}}}(1) & g_{t-1}^{\widehat{\boldsymbol{bg}}}(1) & b_{t-1}^{\widehat{\boldsymbol{bg}}}(1) & 1 \\ r_{t-1}^{\widehat{\boldsymbol{bg}}}(2) & g_{t-1}^{\widehat{\boldsymbol{bg}}}(2) & b_{t-1}^{\widehat{\boldsymbol{bg}}}(2) & 1 \\ \vdots & \vdots & \vdots & \vdots \\ r_{t-1}^{\widehat{\boldsymbol{bg}}}(n) & g_{t-1}^{\widehat{\boldsymbol{bg}}}(n) & b_{t-1}^{\widehat{\boldsymbol{bg}}}(n) & 1 \end{bmatrix} \tag{5.19}$$

$$Y = \begin{bmatrix} r_t^{\boldsymbol{y}}(1) & g_t^{\boldsymbol{y}}(1) & b_t^{\boldsymbol{y}}(1) & 1 \\ r_t^{\boldsymbol{y}}(2) & g_t^{\boldsymbol{y}}(2) & b_t^{\boldsymbol{y}}(2) & 1 \\ \vdots & \vdots & \vdots & \vdots \\ r_t^{\boldsymbol{y}}(n) & g_t^{\boldsymbol{y}}(n) & b_t^{\boldsymbol{y}}(n) & 1 \end{bmatrix} \tag{5.20}$$

n étant le nombre de pixels de l'image. La transformation des pixels d'arrière-plan de t à $t-1$, équation 5.17, peut alors s'écrire matriciellement :

$$Y = XA^T + E \tag{5.21}$$

E est la matrice contenant l'erreur résiduelle pour chaque composante couleur de chaque pixel.

Afin de simplifier les notations, les indices temporels t et $t-1$ ont été omis, notamment pour les matrices X, Y, A et E. Il est entendu que la transformation à estimer est entre les temps $t-1$ et t.

La minimisation par les moindres carrés pondérés consiste à minimiser le coût q en fonction de A :

$$q = \sum_{k=1}^{n} w_t(k) \boldsymbol{e}_t(k)^T \boldsymbol{e}_t(k) \tag{5.22}$$

avec $w_t(k)$ le poids associé au pixel k, scalaire positif ou nul. On peut écrire q matriciellement en fonction de E et d'une matrice diagonale W de taille $n \times n$ contenant les poids $w_t(k)$ pour chaque pixel :

$$q = trace(EWE^T) \tag{5.23}$$

5.4.3.2 Transformation optimale au sens des moindres carrés pondérés

Le problème de minimisation du coût q, équation 5.23 en fonction des paramètres de la transformation linéaire A possède une solution optimale que l'on peut obtenir analytiquement. Il faut pour cela considérer la dérivée du coût en fonction des paramètres, et résoudre l'équation suivante :

$$\frac{\partial q}{\partial A^T :} = 0 \tag{5.24}$$

L'opérateur ":" désigne la concaténation des vecteurs colonnes d'une matrice en un seul long vecteur.

A étant une matrice, et q une trace de matrice, il est nécessaire d'employer les règles de dérivation de matrice. Le calcul s'écrit simplement à condition de respecter les propriétés de linéarité des traces de matrices :

$$\begin{aligned} q &= trace((Y - XA^T)^T W (Y - XA^T)) && (5.25)\\ q &= trace(Y^T W Y) && (5.26)\\ &\quad -trace(Y^T W A X^T)\\ &\quad -trace(A X^T W Y)\\ &\quad +trace(A X^T W X A^T)\\ q &= trace(Y^T W Y) && (5.27)\\ &\quad -2.trace(A X^T W Y)\\ &\quad +trace(A X^T W X A^T)\\ \frac{\partial q}{\partial A^T :} &= -2.(X^T W Y)^T : +2.((X^T W X) A^T)^T : && (5.28) \end{aligned}$$

L'équation 5.24 se réduit donc à :

$$X^T W X A^T = X^T W Y \tag{5.29}$$

La solution optimale au sens des moindres carrés pondérés par W est donnée par :

$$A^T = (X^T W X)^{-1} X^T W Y \tag{5.30}$$

5.4.3.3 Estimation itérative des poids et de la transformation

Les poids de W sont évidemment inconnus *a priori*, et doivent être déterminés conjointement à la transformation A. Pour estimer correctement A, W doit contenir des poids faibles aux pixels correspondant aux objets d'avant-plan, et des poids forts aux pixels d'arrière-plan. La méthode consiste à initialiser W aussi proche que possible de la réalité de façon à obtenir une première estimation de A, puis à réestimer itérativement W.

Si $W^{(i)}$ et $A^{(i)}$ désignent respectivement la matrice de poids et la transformation linéaire estimées à l'itération (i), l'algorithme de minimisation itérative par les moindres carrés pondérés consiste en les 3 étapes suivantes, itérées jusqu'à un critère d'arrêt :

$$\begin{aligned} {A^{(i)}}^T &= (X^T W^{(i)} X)^{-1} X^T W^{(i)} Y && (5.31)\\ E^{(i)} &= Y - X.{A^{(i)}}^T && (5.32)\\ \forall k, W^{(i+1)}[k,k] &= h(E^{(i)}[k]) && (5.33) \end{aligned}$$

La fonction h de l'équation 5.33 met à jour les poids pour l'itération suivante de l'algorithme, en fonction de l'erreur résiduelle obtenue à l'itération courante. Comme la présence

d'un objet d'avant-plan à une position k implique généralement une erreur résiduelle forte en k (équation 5.18), le choix de la fonction h doit être de telle sorte qu'une erreur résiduelle forte entraîne un poids faible.

Un choix possible pour la fonction h est d'utiliser la norme L_2 du résidu :

$$h(\boldsymbol{e}_t(k)) = \frac{1}{1 + \boldsymbol{e}_t(k)^T.\boldsymbol{e}_t(k)} \tag{5.34}$$

Étant donné le nombre important de pixels *outliers*, l'initialisation de $W^{(0)}$ est un problème délicat. On utilisera si possible des informations *a priori* sur la scène, comme par exemple la connaissance de zones de l'image où aucun objet n'est susceptible d'être présent. Dans ce cas, on initialisera à 1 le poids des pixels de cette zone, et à 0 le poids de tous les autres pixels.

5.4.4 Extraction des pixels d'avant-plan à partir de l'arrière plan estimé

Comme précisé précédemment par l'équation 5.18, l'erreur résiduelle $\boldsymbol{e}_t(k)$ contient la somme de l'erreur d'estimation, notée $\boldsymbol{\delta}_t(k)$, et de la contribution des objets d'avant-plan. Les pixels d'avant-plan ont été traités comme *outliers* pendant l'estimation de la transformation de l'arrière-plan. Si cette estimation est correcte, l'erreur résiduelle est beaucoup plus importante aux pixels d'avant-plan qu'aux pixels d'arrière-plan. L'analyse des résidus $\boldsymbol{e}_t(k)$ va alors permettre d'extraire les pixels d'avant-plan de l'image courante $\boldsymbol{y}_t$.

D'autre part, les poids $w_t(k)$, estimés lors de la minimisation itérative par moindre carrés pondérés, apportent une information importante car ils sont faibles aux positions correspondant aux objets d'avant-plan.

5.4.4.1 Analyse des résidus

Afin de séparer les pixels de l'image appartenant à l'avant-plan de ceux appartenant à l'arrière-plan, l'estimation robuste de la distribution des résidus d'arrière-plan est envisagée. Les résidus des pixels d'avant-plan sont considérés ici comme *outliers* par rapport à la distribution, supposée normale, des résidus d'arrière-plan.

Méthodes classiques en estimation robuste de position et dispersion L'estimation robuste de la dispersion de données multivariées et la détection d'*outliers* sont des problèmes classiques en analyse statistique. Notamment, [Mar76] propose des M-estimateurs robustes pour les paramètres de position et de dispersion. [Cam80] détecte les points hors-norme en utilisant la distance de Mahalanobis, calculée à partir de M-estimateurs pour la moyenne et la matrice de covariance. Une autre famille de méthodes consiste à détecter les *outliers* en cherchant les points extrêmes après avoir projeté les données sur certaines directions [Sta81] [Don82]. Une méthode plus géométrique consiste à

calculer l'ellipsoïde de plus petit volume qui englobe au moins la moitié des points [Rou85]. Ces méthodes ont l'inconvénient de ne pas être adaptées pour la suppression d'un nombre important d'*outliers*, et nécessitent un temps de calcul important, dû à la nécessité de rééchantillonner les données ou de les projeter sur un grand nombre de directions.

Estimateur proposé La proportion trop forte d'*outliers* dans nos données ne permet pas d'utiliser ces méthodes classiques. Néanmoins, l'estimation de la transformation A par les moindres carrés pondérés apporte deux informations importantes :

- Tout d'abord, les poids $w_t(k)$ sont faibles pour les pixels k correspondant aux objets d'avant-plan. On les suppose négligeables devant les poids correspondant aux pixels d'arrière-plan.
- D'autre part, les moindres carrés assurent que la distribution des $\sqrt{w_t(k)}.\boldsymbol{e}_t(k)$ a une moyenne nulle.

A partir de ces observations, il est raisonnable de considérer la distribution des $\sqrt{w_t(k)}.\boldsymbol{e}_t(k)$ comme une approximation de la distribution des résidus $\boldsymbol{\delta}_t(k)$ correspondant à l'arrière-plan.

Cette distribution est supposée gaussienne et centrée. Un estimateur de la matrice de covariance des résidus d'arrière-plan est :

$$\widehat{\Gamma} = \frac{1}{\sum_{k=1}^{n} w_t(k)} \sum_{k=1}^{n} w_t(k).\boldsymbol{e}_t(k).\boldsymbol{e}_t(k)^T \tag{5.35}$$

$\widehat{\Gamma}$ est une matrice de covariance pondérée. La contribution des pixels d'arrière-plan est plus forte que celle des pixels d'avant-plan lors de son estimation. A partir de cette modélisation de la distribution des résidus $\boldsymbol{\delta}_t(k)$, il est possible de détecter quels résidus $\boldsymbol{e}_t(k)$ proviennent de pixels d'avant-plan. Ils sont en effet considérés comme étant *outliers* de la distribution des $\boldsymbol{\delta}_t(k)$. La distance de Mahalanobis des résidus à leur distribution est alors considérée :

$$m_t(k)^2 = \boldsymbol{e}_t(k)^T \widehat{\Gamma}^{-1} \boldsymbol{e}_t(k) \tag{5.36}$$

Pour des données multivariées de dimension p (ici $p = 4$), distribuées normalement, les distances au carré $m_t(k)^2$ sont les réalisations d'une distribution χ^2 à p degrés de liberté, notée χ^2_p. Les *outliers* sont alors les résidus pour lesquelles la distance est grande, et n'est pas incluse dans un certain intervalle de confiance.

On peut calculer une valeur limite de la distance correspondant à un certain intervalle de confiance. Pour une probabilité c de bonne détection d'un résidu d'arrière-plan, on choisira comme seuil $F_p^{-1}(c)$, F_p^{-1} étant l'inverse de la fonction de répartition de la distribution χ^2_p. La fonction F_p est définie comme suit :

$$F_p(c) = \frac{\gamma_{\frac{c}{2}}\left(\frac{p}{2}\right)}{\gamma\left(\frac{p}{2}\right)} \tag{5.37}$$

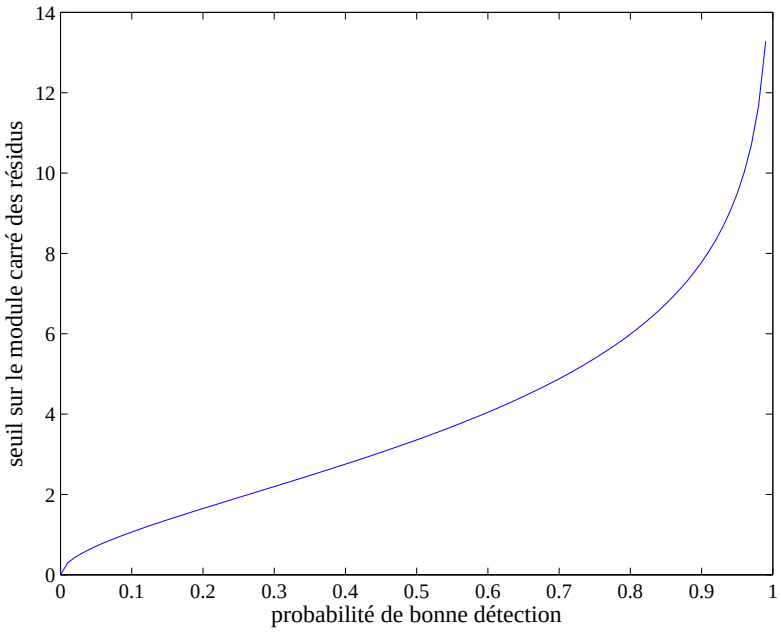


FIG. 5.17 – Inverse de la fonction de répartition de la distribution χ^2_4

avec $\gamma(a) = \int_0^\infty t^{a-1}e^{-t}dt$ et $\gamma_x(a) = \int_0^x t^{a-1}e^{-t}dt$

Pour des données de dimension $p = 4$ (les 3 composantes couleur et la constante 1), et une probabilité de bonne détection de 95% des points d'arrière plan, la valeur du seuil à choisir est $F_4^{-1}(0.95) = 9.49$ (figure 5.17).

5.4.5 Extension aux vecteurs de caractéristiques

La méthode de séparation de pixels d'avant-plan présentée jusqu'ici permet d'estimer, à partir d'une approximation de l'arrière-plan au temps $t-1$, quels pixels au temps t appartiennent à l'avant-plan, en faisant l'hypothèse que l'arrière-plan suit globalement une transformation linéaire des couleurs. Nous avons vu précédemment que l'arrière-plan subit des variations qui ne sont pas modélisables directement par une transformation linéaire des pixels. Par exemple, les mouvements faibles de feuillage, ou du véhicule lors de la montée des passagers sont des changements qui d'une part sont locaux, et d'autre part dépendent des pixels voisins. En réalité, la transformation linéaire globale des pixels de l'arrière-plan parvient à modéliser uniquement les variations dues aux contrôles automatiques de la caméra.

Néanmoins, le formalisme explicité précédemment peut s'étendre à d'autres caractéristiques de l'image que la valeur des pixels. On peut effectivement chercher une représentation de l'image pour laquelle la transformation de l'arrière-plan se rapproche d'une transformation linéaire, et ainsi prendre en compte les variations non linéaires des pixels. Il s'agit donc

de définir un vecteur de caractéristiques locales de l'image, pour chaque pixel k, tel que la variation temporelle des vecteurs correspondant à l'arrière-plan soit modélisable de façon raisonnable par une transformation linéaire.

De manière similaire à l'équation 5.15, connaissant une estimation des vecteurs de caractéristiques de l'arrière-plan $\boldsymbol{x}_{t-1}(k)$ et les vecteurs de caractéristiques de l'image observée $\boldsymbol{y}_t(k)$, nous cherchons la transformation linéaire A_t des pixels l'arrière-plan :

$$\forall k, \boldsymbol{y}_t(k) = A_t \boldsymbol{x}_{t-1}(k) + \boldsymbol{e}_t(k) \tag{5.38}$$

5.4.5.1 Vecteur du voisinage d'un pixel

Une première solution consiste à prendre en compte dans le vecteur de caractéristiques le voisinage de chaque pixel. On définit alors $\boldsymbol{y}_t(k)$ comme l'ensemble des pixels au temps t qui sont inclus dans le voisinage du pixel k. Par exemple, en considérant une fenêtre W_k autour du pixel k, le vecteur suivant peut être défini :

$$\boldsymbol{y}_t(k) = \{(r_t(k'), g_t(k'), b_t(k')), k' \in W_k\} \tag{5.39}$$

Dans ce cas, la transformation linéaire prend toujours en compte les variations globales en luminance et chrominance, mais elle est surtout capable de modéliser les petits mouvements globaux de la scène, qui apparaissent dans A_t comme une permutation.

Les principales limitations de ce type de vecteur apparaissent clairement :

- Les mouvements qui peuvent être modélisés par A sont uniquement des mouvements globaux. Or dans le type de séquences que nous traitons, les mouvements de l'arrière-plan sont souvent locaux. L'exemple le plus parlant est peut être celui de la montée des passagers, qui provoque un déplacement vertical d'uniquement une partie de l'arrière-plan (la partie vitrée).
- Les mouvements globaux modélisables sont d'ailleurs réduits aux seules translations.
- Pour prendre en compte de manière réaliste des mouvements globaux, la taille de la fenêtre W doit être conséquente, et la dimension du vecteur de caractéristiques devient un problème lorsque les ressources en mémoire sont limitées.

5.4.5.2 Vecteur de statistiques sur la distribution locale des couleurs

Une solution alternative consiste alors à conserver le modèle de transformation linéaire pour les changements globaux de couleurs, mais en utilisant des vecteurs de caractéristiques qui sont eux-même robustes aux petits mouvements locaux de l'arrière-plan.

Un mouvement local léger peut modifier radicalement les valeurs de chaque pixel, par exemple dans le cas où la scène est très texturée. Par contre, la distribution des couleurs autour de chaque pixel s'en voit très peu modifiée. En effet, la texture d'un objet de la scène varie généralement très peu d'une position spatiale donnée à une position voisine. Si l'on considère par exemple un arbre dont les branches bougent à cause du vent, la texture à une position donnée de l'arbre restera relativement stable : la proportion des couleurs

sera relativement constante. C'est ce qui nous incite à considérer cette distribution locale des couleurs comme caractéristique de l'image pour chaque pixel.

Une manière directe de représenter la distribution des couleurs autour d'un pixel est de construire l'histogramme couleur de son voisinage. Intervient alors la question du stockage de cet histogramme. Il n'est évidemment pas envisageable de conserver l'histogramme couleur autour de chaque pixel. Avec 3 composantes couleurs et un nombre de *bins* raisonnable de 16 par composante, la dimension du vecteur de caractéristique serait de $16^3 = 4096$.

De façon plus réaliste, nous allons représenter la distribution des couleurs par des mesures statistiques, qui ont l'avantage de former une représentation compacte de l'information, tout en restant suffisamment discriminantes pour des distributions différentes. Les statistiques considérées sont les moyennes $\bar{r}(k)$, $\bar{g}(k)$, $\bar{b}(k)$ de chaque composante ainsi que les moments centrés multivariés jusqu'à un certain ordre m, localement à la fenêtre W_k :

$$\mu_{\alpha,\beta,\gamma}(k) = \frac{1}{Card(W_k)} \sum_{k' \in W_k} (r(k') - \bar{r}(k))^\alpha (g(k') - \bar{g}(k))^\beta (b(k') - \bar{b}(k))^\gamma \tag{5.40}$$

avec $2 \leq \alpha + \beta + \gamma \leq m$.

A l'ordre $m = 2$, on obtient le vecteur de dimension $p = 10$ suivant (l'indice temporel t a été volontairement omis pour des soucis de clarté) :

$$\boldsymbol{y}(k) = \begin{bmatrix} \bar{r}(k) \\ \bar{g}(k) \\ \bar{b}(k) \\ \mu_{2,0,0}(k) \\ \mu_{0,2,0}(k) \\ \mu_{0,0,2}(k) \\ \mu_{1,1,0}(k) \\ \mu_{0,1,1}(k) \\ \mu_{1,0,1}(k) \\ 1 \end{bmatrix} \tag{5.41}$$

Notons à nouveau la présence de la constante 1 en fin du vecteur, permettant de modéliser par A_t l'ajout global d'un vecteur constant entre $t-1$ et t.

5.4.5.3 Pertinence d'une transformation linéaire des moments statistiques

Une question se pose à propos de la pertinence d'estimer une transformation linéaire globale sur des moments locaux. Effectivement, il est difficile d'imaginer une quelconque transformation réelle qui transformerait les moments d'ordre supérieur à 2 de manière linéaire. En réalité, le vecteur de caractéristiques a été introduit au modèle d'arrière-plan afin d'apporter de la robustesse par rapport aux changements non linéaires, et les moments centrés locaux sont justement des caractéristiques robustes à ces changements. On peut

alors s'attendre à une contribution faible des moments d'ordre ≥ 2 en ce qui concerne l'estimation de A_t, et se demander quel est l'intérêt d'estimer une transformation linéaire à $m.(m-1)$ degrés de liberté, lorsque seuls les paramètres concernant les moyennes de chaque composante ont un réel sens physique.

L'intérêt d'inclure les moments centrés d'ordre ≥ 2 dans l'estimation de A_t est bien réel. Pour s'en persuader, il faut se rapporter à l'étape de détection des pixels d'avant-plan par la distance de Mahalanobis (équation 5.36).

En plus d'apporter une robustesse aux petites variations en mouvement, les statistiques permettent de discriminer aisément des zones texturées différemment, même lorsque leur couleur moyenne est quasiment identique. Effectivement, en supposant par exemple que la transformation A_t estimée soit l'identité pour les moments d'ordre ≥ 2, le pixel d'un objet texturé d'avant-plan de même couleur que l'arrière-plan sera détecté car son erreur aura une distance de Mahalanobis anormalement grande. La matrice de covariance des résidus d'arrière-plan $\widehat{\Gamma}$ (équation 5.35) modélise par ailleurs les variations acceptables des statistiques entre les temps $t-1$ et t.

Multilinéarité des moments statistiques Une remarque importante, et qui nous conforte dans ce choix du vecteur de caractéristiques, est que les moments statistiques ont la propriété de multilinéarité. Lorsque les données subissent une transformation linéaire de t à $t-1$, ce qui est le cas d'après nos hypothèses, les moments statistiques s'exprime de t aussi linéairement en fonction des moments statistiques de $t-1$.

Normalisation du vecteur de caractéristiques Une autre remarque à prendre en compte concerne la dynamique des variables à l'intérieur du vecteur de caractéristiques. L'estimation de la transformation linéaire A_t (équation 5.38) s'effectue en minimisant la somme pondérée sur k des modules des résidus $\boldsymbol{e}_t(k)$. Cela implique qu'une variable à la dynamique trop importante par rapport aux autres sera privilégiée lors de la minimisation, car elle contribuera beaucoup plus au module du résidu que les autres variables.

Les moments statistiques du vecteur de caractéristiques sont donc normalisés afin que leur contribution pendant la minimisation soit à peu près équivalente.

Détection des points d'avant-plan Afin de détecter les points d'avant-plan à partir des résidus, il faut à nouveau considérer la distance de Mahalanobis des résidus à la distribution des résidus d'arrière-plan (équation 5.36). Avec ce choix de vecteur de caractéristiques, les distances suivent cette fois une distribution χ^2_p à $p = 10$ degrés de liberté. Pour un taux de bonnes détections de 95% des points d'arrière-plan, la valeur du seuil à choisir est $F^{-1}_{10}(0.95) = 18.31$ (figure 5.18).

5.4.6 Hypothèses multiples sur l'état de l'arrière-plan

Nous avons jusqu'à maintenant travaillé à extraire les pixels d'avant-plan dans une image de la séquence vidéo à l'instant t, à partir de cette image observée et d'une estimation

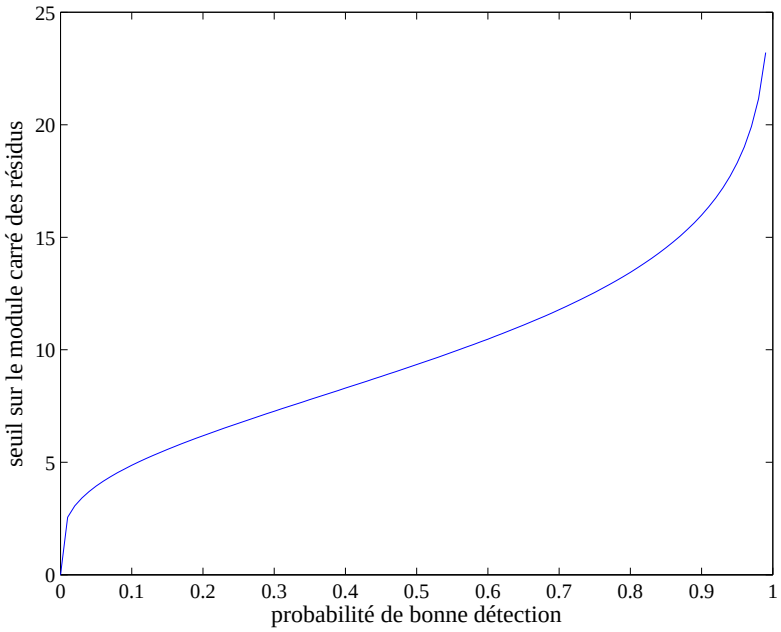


FIG. 5.18 – Inverse de la fonction caractéristique de la distribution χ^2_{10}

$\widehat{\boldsymbol{bg}_{t-1}}$ de l'arrière-plan au temps précédent. Il est maintenant nécessaire de comprendre comment cette estimation de l'arrière-plan est obtenue, afin d'avoir une vision complète de l'algorithme d'extraction de pixels d'avant-plan sur une séquence vidéo entière.

5.4.6.1 Besoin d'hypothèses multiples

Dans notre contexte de caméra de surveillance embarquée dans un véhicule de transport en commun, lors d'un arrêt, des passagers sont *a priori* visibles pendant toute la séquence, de l'ouverture à la fermeture des portes. En particulier, il n'est pas possible de considérer que la vue n'est composée que de pixels d'arrière-plan au début de la séquence. A une position donnée dans l'image, les passagers peuvent apparaître plus longtemps que l'arrière-plan, et peuvent rester longtemps statiques, par exemple lorsqu'un passager achète un titre de transport. Il est nécessaire que l'algorithme prenne en compte différentes hypothèses d'arrière-plan, quitte à décider plus tard dans la séquence de l'hypothèse la plus probable, lorsque de nouvelles informations sont disponibles. Un cadre à hypothèses multiples permet de passer d'une estimation de fond à une autre, à chaque fois que l'image nouvellement observée rend l'ancienne hypothèse de fond moins probable qu'une autre.

Nous ne nous situons pas dans un contexte où l'arrière-plan peut *a priori* avoir plusieurs états, comme ce qui est par exemple supposé dans des méthodes de soustraction de fond basées sur les mélanges de Gaussiennes. En effet les séquences vidéo d'arrêts sont courtes, et les variations de l'arrière-plan sont prises en compte à un niveau plus bas-niveau de l'algorithme, par l'estimation de la transformation linéaire et l'utilisation de vecteurs de

caractéristiques robustes.

La méthode proposée fait au contraire la supposition d'un unique état de fond, mais utilise plusieurs hypothèses, dont une seule est vraie, les autres représentant en réalité des objets d'avant-plan cachant le fond.

5.4.6.2 Lien entre vecteurs de caractéristiques et hypothèses

L'algorithme considère un ensemble de H hypothèses pour chaque position spatiale k. Chacune de ces hypothèses est basiquement composée d'un vecteur de caractéristiques et de la probabilité que ce vecteur soit celui de l'arrière-plan. Dans la pratique on choisit $H = 5$. Une hypothèse sera utilisée pour l'arrière-plan tandis que les autres seront utilisées pour les différents états d'avant-plan.

On notera $\boldsymbol{x}_t^{\{1\}}(k), \ldots, \boldsymbol{x}_t^{\{H\}}(k)$ les vecteurs de caractéristiques correspondant aux H hypothèses pour le pixel k au temps t, et $p_t^{\{1\}}(k), \ldots, p_t^{\{H\}}(k)$ leurs probabilités associées.

Le vecteur de caractéristiques estimé comme arrière-plan au temps $t-1$ et au pixel k est simplement celui dont la probabilité d'arrière-plan est la plus élevée. Ce vecteur $\widehat{\boldsymbol{bg}_{t-1}}(k)$ est alors utilisée pour l'estimation de la transformation linéaire A_t et la détection des pixels d'avant-plan, comme expliqué précédemment.

Mise à jour des vecteurs de caractéristiques Chaque vecteur de caractéristiques observé $\boldsymbol{y}_t(k)$ est assigné à l'une des hypothèses d'arrière-plan pour le pixel k, en fonction d'une mesure de distance entre $\boldsymbol{y}_t(k)$ et chacun des $\boldsymbol{x}_t^{\{h\}}(k)$. On utilise pour cela la transformation A_t estimée et la distance de Mahalanobis à la distribution des résidus (équation 5.36).

Dénotons par h_0 l'hypothèse pour laquelle la distance est la plus faible. Il y a alors deux cas possibles :

1. La distance est inférieure au seuil fixé, et l'observation correspond bien à l'hypothèse h_0. Cette hypothèse est alors directement mise à jour par l'observation :

$$\boldsymbol{x}_t^{\{h_0\}}(k) = \boldsymbol{y}_t(k) \tag{5.42}$$

2. La distance est supérieure au seuil fixé, et l'observation ne correspond à aucune des hypothèses existantes. Une nouvelle hypothèse est alors créée avec l'observation. Comme on souhaite conserver un nombre fixe d'hypothèses, la nouvelle hypothèse remplace l'hypothèse existante h_- la moins probable :

$$\boldsymbol{x}_t^{\{h_-\}}(k) = \boldsymbol{y}_t(k) \tag{5.43}$$

Les autres hypothèses d'arrière-plan ne sont pas observées, mais doivent aussi être mises à jour, de façon à prendre en compte les transformations globales de l'arrière-plan. En effet, considérons une position k de l'image, et les temps t_1 et t_2 avec $t_1 < t_2$ pendant lesquels un objet cache l'arrière-plan. Celui-ci subit les transformations $A_{t_1}, A_{t_1+1}, \ldots, A_{t_2}$, si bien que la transformation totale entre les temps t_1 et t_2 est :

$$A_{t_1,t_2} = A_{t_2} A_{t_2-1} \dots A_{t_1} \tag{5.44}$$

Lorsque l'objet disparaît et que l'arrière-plan est de nouveau observé au temps t_2, l'algorithme doit être capable d'associer $\boldsymbol{y}_{t_2}(k)$ et l'hypothèse $\boldsymbol{x}_{t_1-1}^{\{h\}}(k)$. Pour que la distance entre ces deux vecteurs soit minimale, il est nécessaire de prédire l'hypothèse h par A_t à chaque instant t ou elle n'est pas observée. De cette façon, on a bien $\boldsymbol{y}_{t_2}(k) = A_{t_1,t_2}\boldsymbol{x}_{t_1-1}^{\{h\}}(k)+\boldsymbol{e}_{t_2}(k)$.

La mise à jour des hypothèses d'arrière-plan non observées est alors :

$$\boldsymbol{x}_t^{\{h\}}(k) = A_t \boldsymbol{x}_{t-1}^{\{h\}}(k) \tag{5.45}$$

Mise à jour des probabilités d'arrière-plan Les probabilités d'appartenance à l'arrière-plan $p_t^{\{h\}}(k)$ doivent aussi être mises à jour, en fonction de la fréquence d'apparition de chaque hypothèse. Si l'on désigne par $V_t^{\{h\}}(k)$ le nombre de fois où l'hypothèse h au pixel k a été observée du début de la séquence jusqu'au temps t, la mise à jour de la probabilité d'arrière-plan est régie par :

$$p_t^{\{h\}}(k) = \frac{1}{\sum_{g=1}^{H} V_t^{\{g\}}} V_t^{\{h\}}(k) \tag{5.46}$$

Cette mise à jour des probabilités suppose que l'arrière-plan est l'élément de la scène qui est *stable* plus longtemps que les autres objets pendant la séquence. Par stable, on sous-entend ici que les pixels d'arrière-plan suivent la transformation globale estimée et présentent des variations auxquelles le vecteur de caractéristiques proposé est robuste.

5.4.7 Évaluation des performances

Les performances de l'algorithme proposé sont évaluées quantitativement sur des données synthétiques, afin de disposer d'une vérité terrain. Une séquence vidéo synthétique est construite à partir d'une photo $\boldsymbol{z}$, faisant office d'arrière plan, sur laquelle on simule des variations correspondant aux conditions de notre contexte. Plus précisément, il y a deux types de variations simulées :

1. Des mouvements locaux sur toute la surface de l'image sont générés par des petits déplacements de chaque pixel, régie par une sinusoïde dont la phase dépend du temps :

$$\boldsymbol{z}'_t(x,y) = \boldsymbol{z}(x + a\sin(\omega x + \phi t), y + a\sin(\omega y + \phi t)) \tag{5.47}$$

   avec α la constante contrôlant l'amplitude du déplacement, ϕ la vitesse d'oscillation et ω la fréquence spatiale des oscillations.

2. Les contrôles automatiques de gain et de balance des couleurs sont simulés en appliquant une transformation linéaire globalement sur tous les pixels. La transformation linéaire est la combinaison d'une rotation des teintes et d'une variation de l'intensité, et est appliquée dans l'espace de couleur YUV. Les paramètres dépendent du temps :



(a) (b)

FIG. 5.19 – Estimation de l'arrière-plan par la méthode présentée. (a) : Image originale. (b) :Arrière-plan estimé

$$A_t = \begin{bmatrix} \alpha(t) & 0 & 0 \\ 0 & \cos(\theta(t)) & \sin(\theta(t)) \\ 0 & -\sin(\theta(t)) & \cos(\theta(t)) \end{bmatrix} \tag{5.48}$$

$$\boldsymbol{z}_t''(x,y) = A_t \boldsymbol{z}'(x,y) \tag{5.49}$$

On superpose à $\boldsymbol{z}_t''$ la vidéo d'une personne prise sur fond bleu et extraite au préalable par un seuillage simple. Le résultat est une séquence vidéo comprenant un arrière-plan dynamique et une personne mobile. Comme l'assemblage entre arrière-plan et avant-plan est artificiel, la vérité terrain est connue, ce qui permet une évaluation quantitative des performances.

5.4.7.1 Évaluation de l'estimation de l'arrière plan

Il est possible, avec le modèle d'arrière-plan présenté ici, d'estimer une *image* de l'arrière-plan. Ceci est différent de l'estimation de l'arrière-plan que l'on conserve tout au long de l'algorithme, et qui consiste en un champ de vecteurs de caractéristiques de dimension 10. Afin d'obtenir notre estimation de l'image d'arrière-plan, on conserve pour chaque vecteur d'hypothèse d'arrière-plan le pixel du centre de la fenêtre d'analyse correspondante. Ce pixel est mis à jour par transformation linéaire de la même manière que les vecteurs de caractéristiques lorsque l'arrière-plan est caché par un objet, de façon à ce que l'image d'arrière-plan estimée suive les variations globales de couleur.

Afin d'évaluer la qualité de l'arrière-plan estimé, l'erreur quadratique moyenne est mesurée entre l'arrière-plan estimé et l'arrière-plan de la séquence vidéo de synthèse. La méthode d'estimation d'arrière-plan présentée est aussi comparée avec la méthode plus classique du filtrage médian temporel. Ce filtrage consiste à choisir comme estimation d'arrière-plan la valeur médiane de toutes les valeurs que prend chaque pixel sur la séquence



FIG. 5.20 – Estimation de l'arrière-plan par filtrage médian temporel

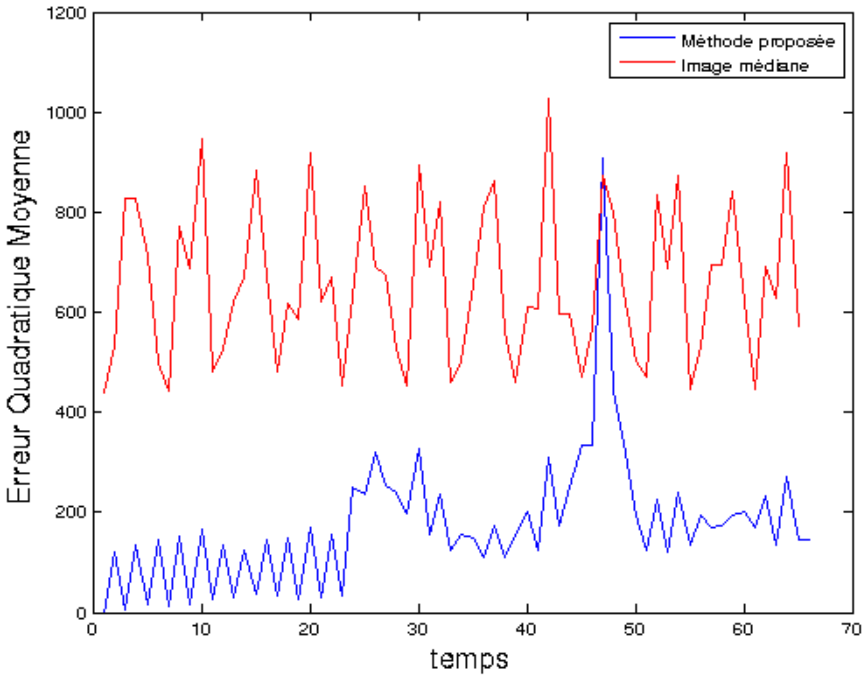


FIG. 5.21 – Erreur quadratique moyenne entre arrière-plan réel et estimé. Comparaison avec l'estimation par image médiane

Modèle d'arrière plan utilisé	Taux de bonnes détection	Taux de fausses détection
Modèle proposé	87%	6%
Mélanges de Gaussiennes	73%	35%

Tab. 5.1 – Comparaison des taux de détection entre l'estimation de fond proposée et l'estimation de fond par mélanges de Gaussiennes

entière. L'image d'arrière-plan obtenue par filtrage médian est présentée figure 5.20. L'erreur quadratique moyenne en fonction du temps est présentée figure 5.21 pour les deux estimateurs. Notre méthode permet en moyenne une meilleure estimation de l'arrière-plan que la méthode du filtrage médian temporel.

5.4.7.2 Comparaison quantitative avec les mélanges de Gaussiennes

Ce type de séquence vidéo synthétique permet de comparer l'algorithme proposé avec d'autres algorithmes de soustraction de fond. Une méthode classique de soustraction de fond consiste à modéliser la distribution des valeurs prises temporellement en chaque pixel par une somme pondérée de Gaussiennes, dont les paramètres sont appris par un algorithme de type *Expectation-Maximization* (EM) [GS99].

Les figures 5.23 et 5.24 présentent les résultats en bonnes détections et fausses détections sur la séquence de synthèse, pour l'algorithme d'estimation de fond proposé et le modèle de mélanges de Gaussiennes. En moyenne, 87% des points d'avant-plan sont correctement détectés avec la méthode proposée, et 6% des points détectés comme avant-plan sont des fausses détections. Pour comparaison, la méthode utilisant des mélanges de Gaussiennes parvient à détecter correctement 73% des points d'avant-plan, avec 35% de fausses détections.

Un récapitulatif de ces résultats est présenté dans le tableau 5.1. Un exemple de résultat de détection comparant les deux méthodes est illustré figure 5.22.

5.4.7.3 Performances sur séquences réelles

L'algorithme est bien entendu évalué sur des séquences vidéos réelles de caméras embarquées dans un véhicule de transports en commun. La quantification des performances est délicate pour les séquences réelles puisque nous ne possédons pas de vérité terrain. Les résultats quantitatifs seront présentés pour la détection de personne, qui est un module plus haut-niveau et pour lequel la construction d'une vérité terrain peut être envisagée. Les résultats qualitatifs sont néanmoins présentés figure 5.25.

Il est clair que le mélange de Gaussiennes, à cause de sa lenteur d'apprentissage et du fait qu'il ne prend pas en compte l'information spatiale, ne parvient pas à être robuste aux différentes variations de l'arrière-plan. La méthode proposée parvient quant-à-elle à séparer correctement l'avant-plan et l'arrière-plan, dès les premières images de la séquence.

Notons par ailleurs certaines limitations de l'algorithme proposé. Les reflets qui apparaissent dans certaines parties vitrées du véhicule et dans les miroirs sont détectés comme



séquence originale algorithme proposé mélange de Gaussiennes

FIG. 5.22 – Détection de pixels d'avant-plan sur la séquence synthétique

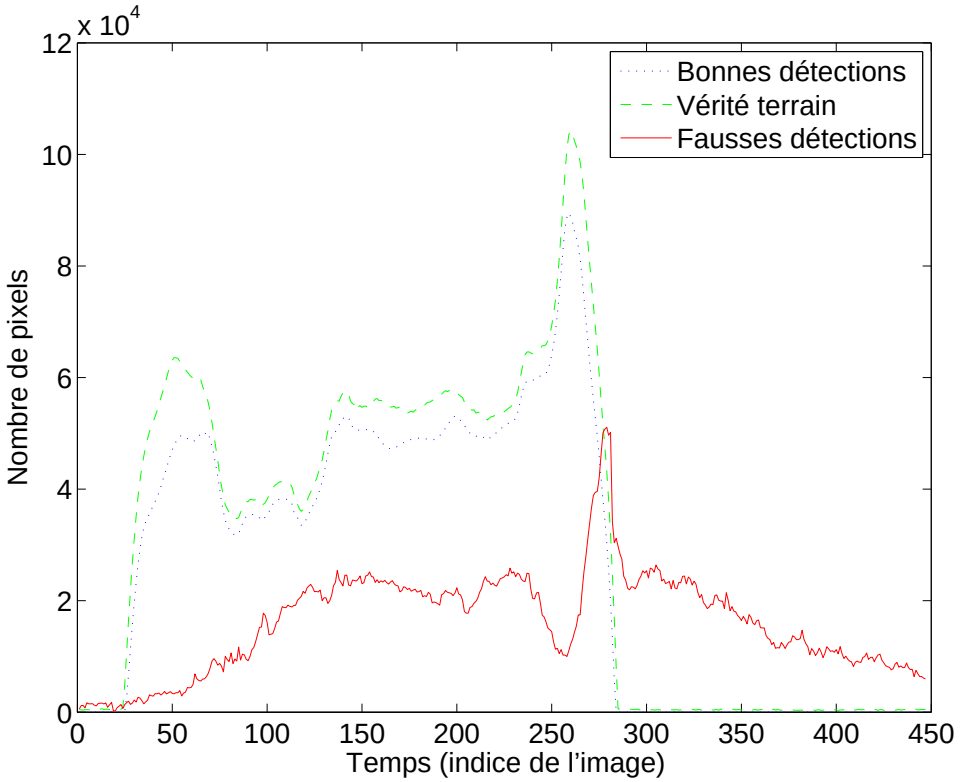


FIG. 5.23 – Performance en bonnes détections et fausses détections de l'estimation de fond proposée sur la séquence synthétique

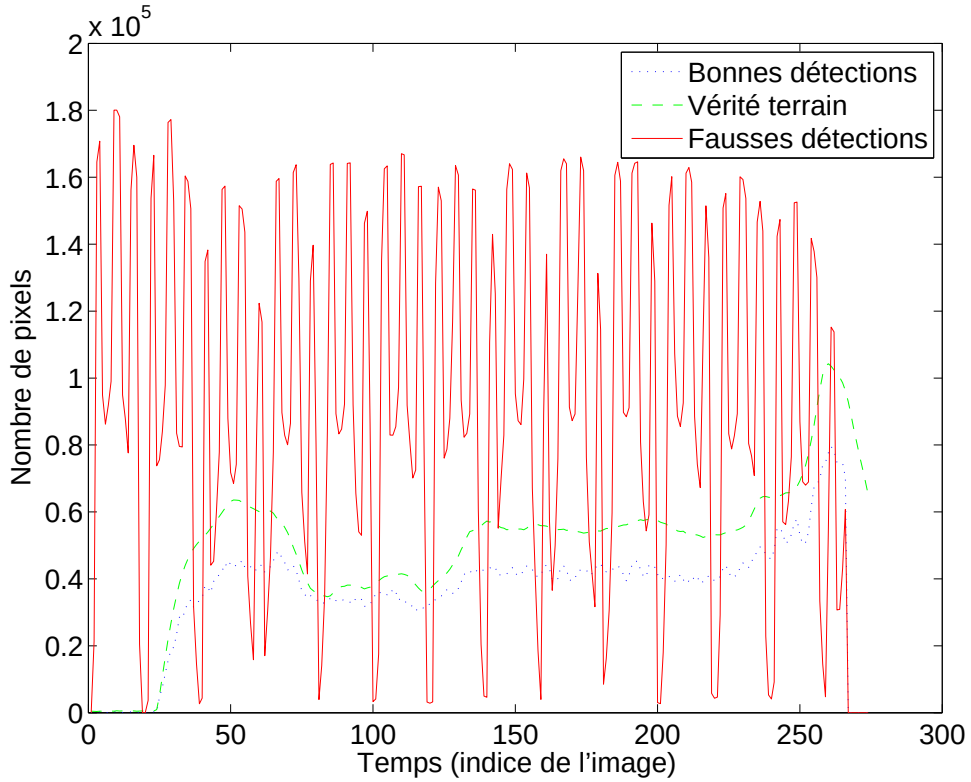


FIG. 5.24 – Performance en bonnes détections et fausses détections de l'estimation de fond par mélange de Gaussiennes sur la séquence synthétique



image originale algorithme proposé mélange de Gaussiennes

FIG. 5.25 – Détection de pixels d'avant-plan sur scène réelle

appartenant à l'avant-plan, même lorsque le reflet semble assez léger. D'autre part, lorsqu'un objet cache l'arrière-plan pendant longtemps et que les variations globales sont importantes, on assiste à une dérive de la prédiction par A_t (équation 5.45), due à l'accumulation des erreurs de prédiction. Le résultat est que lorsque l'objet disparaît et laisse place à l'arrière-plan, l'association entre l'observation courante et l'hypothèse de l'arrière-plan ne peut être faite, ce qui fait chuter la probabilité d'arrière-plan de l'hypothèse qui est réellement l'arrière-plan.

5.4.8 Implémentation efficace du calcul des vecteurs

L'introduction des vecteurs de statistiques sur la distribution locale des couleur apporte une meilleure robustesse aux variations locales de la séquence d'arrêt du véhicule, par rapport à un vecteur contenant simplement les composantes couleur de chaque pixel. Ces vecteurs permettent aussi une meilleure discrimination de zones aux couleurs similaires mais texturées différemment, utile lorsqu'un objet d'avant-plan possède des couleurs proches de celles de l'arrière-plan.

Ces avantages sont au prix d'une occupation mémoire plus importante et d'un temps de calcul supplémentaire dû à la nécessité de calculer les moments statistiques localement pour chaque pixel. Ce calcul de moments statistiques est d'ailleurs beaucoup trop lourd pour être réalisé de manière naïve, si l'on désire une détection en temps-réel des pixels d'avant-plan. Il est proposé ici une méthode de calcul rapide des moments statistiques, prenant en compte la redondance spatiale et la forme rectangulaire des fenêtres d'analyse.

5.4.8.1 Expression des moments centrés multivariés

L'optimisation de calcul proposée nécessite d'exprimer les moments statistiques centrés multivariés en fonction de sommes de puissances définies comme suit :

$$P_{u,v,w}(k) = \sum_{k' \in W_k} r(k')^u g(k')^v b(k')^w \tag{5.50}$$

Pour l'ordre 2, les variances et covariances peuvent s'écrire :

$$\mu_{2,0,0}(k) = \frac{1}{Card(W_k)} P_{2,0,0}(k) - \frac{1}{Card(W_k)^2} P_{1,0,0}(k)^2 \tag{5.51}$$

$$\mu_{1,1,0}(k) = \frac{1}{Card(W_k)} P_{1,1,0}(k) - \frac{1}{Card(W_k)^2} P_{1,0,0}(k) P_{0,1,0}(k) \tag{5.52}$$

$$\vdots$$

5.4.8.2 Calcul rapide des sommes de puissances

Pour une image donnée, les sommes de puissances $P_{u,v,w}(k)$ dans une fenêtre rectangulaire W_k de taille quelconque peuvent être calculées en temps constant avec très peu

d'opérations, à condition de connaître les sommes de puissances $R_{u,v,w}(x,y)$ de toutes les fenêtres rectangulaires dont le coin haut-gauche est fixé au coin haut-gauche de l'image et le coin bas-droite aux coordonnées (x,y) :

$$R_{u,v,w}(x,y) = \sum_{x'=0,y'=0}^{x,y} r(x',y')^u g(x',y')^v b(x',y')^w \tag{5.53}$$

L'indice de position spatial k dans l'image a été remplacé ici par l'équivalent (x,y) dénotant les coordonnées du pixel k. Soit (x_1,y_1) et (x_2,y_2) les coordonnées des coins haut-gauche et bas-droite de la fenêtre W_k. Il est simple d'écrire $P_{u,v,w}(k)$ en fonction de $R_{u,v,w}$:

$$\begin{aligned} P_{u,v,w}(k) &= R_{u,v,w}(x_2,y_2) + R_{u,v,w}(x_1-1,y_1-1) \\ &\quad -R_{u,v,w}(x_2,y_1-1) - R_{u,v,w}(x_1-1,y_2) \end{aligned} \tag{5.54}$$

Par conséquent, lorsque les valeurs $R_{u,v,w}(x,y)$ sont connues pour chaque point (x,y) de l'image et pour tout u,v,w tels que $u+v+w \leq m$, le calcul de $\mu_{\alpha,\beta,\gamma}$ en toute fenêtre rectangulaire se fait directement en combinant les équations 5.51, 5.52 et 5.54. Le précalcul des $R_{u,v,w}(x,y)$ est réalisable en temps linéaire par rapport au nombre de pixels, grâce à la formule récursive suivante :

$$\begin{cases} R_{u,v,w}(0,0) &= r(0,0)^u g(0,0)^v b(0,0)^w \\ R_{u,v,w}(0,y) &= R_{u,v,w}(0,y-1) + r(0,y)^u g(0,y)^v b(0,y)^w \\ R_{u,v,w}(x,0) &= R_{u,v,w}(x-1,0) + r(x,0)^u g(x,0)^v b(x,0)^w \\ R_{u,v,w}(x,y) &= R_{u,v,w}(x-1,y) + R_{u,v,w}(x,y-1) \\ & \quad -R_{u,v,w}(x-1,y-1) + r(x,y)^u g(x,y)^v b(x,y)^w \end{cases}$$

5.4.8.3 Considérations sur l'espace mémoire

A l'ordre $m=2$, l'optimisation proposée nécessite le précalcul et le stockage des valeurs de $R_{1,0,0}$, $R_{0,1,0}$, $R_{0,0,1}$, $R_{2,0,0}$, $R_{0,2,0}$, $R_{0,0,2}$, $R_{1,1,0}$, $R_{1,0,1}$ et $R_{0,1,1}$. Cela correspond à un espace mémoire de $9.n$ avec n le nombre de pixels de l'image, soit environ 22 mégaoctets pour une image en résolution 640×480 et des entiers sur 8 octets.

5.4.9 Conclusions sur l'estimation du fond à un arrêt

Une méthode d'estimation de l'arrière-plan a été présentée pour le cas des séquences d'arrêt du véhicule. Les variations de la scène en mouvement et en illumination nous ont incité à développer un algorithme robuste, basé sur une modélisation de chaque voisinage par une distribution de couleur. L'algorithme proposé est proche des méthodes d'estimation de fond dites *prédictives*, dans le sens où un recalage global et robuste des variations d'illumination entre deux temps consécutif est réalisé. Il est aussi proches des méthodes

non-prédictives telles que les mélanges de Gaussiennes, par sa prise en compte d'hypothèses multiples pour l'état courant de l'arrière-plan, sans distinction sur leur ordre temporel d'apparition. Comme le montrent les résultats obtenus sur les séquences réelles et synthétiques, l'estimation de l'arrière-plan et la détection associée des pixels d'avant-plan est performante dans notre contexte, pour lequel des méthodes classiques comme les mélanges de Gaussiennes ne sont pas adaptées. Elle nécessite par contre des calculs lourds, ce qui lui empêche de fonctionner en temps réel. Malgré l'optimisation proposée pour le calcul des statistiques sur les distributions de couleurs, la cadence de traitement est d'environ une image par seconde sur un processeur 2.4Ghz et pour des images de 640×480 pixels. Il est toutefois à noter que notre contexte n'impose pas le fonctionnement en temps réel, bien que celui-ci aurait été bienvenu. Les séquences d'arrêt peuvent être traitées hors-ligne, pendant une relecture des bandes vidéo enregistrées, sur un ordinateur plus puissant.

5.5 Conclusions sur l'extraction d'informations bas-niveau

Ce chapitre a été l'objet de méthodes permettant l'extraction d'informations bas-niveau, caractérisant chaque pixel de la vidéo par rapport aux classes recherchées. Nous avons ainsi présenté la méthode mise en œuvre pour l'extraction des pixels de teinte chair, basée sur un apprentissage statistique de la teinte chair. La détection des pixels appartenant à des zones vitrées a aussi été étudiée, ainsi que l'estimation de la position de la porte. Enfin une méthode d'estimation de l'arrière-plan et l'algorithme associé de détection des pixels d'avant-plan ont été développés et validés sur des séquences synthétiques et réelles. Ces différentes caractéristiques extraites des données vidéos sont représentées sous la forme de cartes de probabilités, ce qui va permettre une combinaison naturelle de ces diverses sources d'information, que ce soit pour la détection de personnes comme présentée dans le chapitre suivant, ou pour d'autres applications annexes comme la compression sélective, introduite plus tard dans le chapitre 7.

Chapitre 6

Détection et suivi de personnes

Sommaire

6.1 Modèle de visage (fusion au niveau pixel) 136
6.1.1 Fusion teinte chair et mouvement 137
6.1.2 Modèle d'ellipse de peau 137
6.1.2.1 Paramétrisation des ellipses 138
6.1.2.2 Analogie avec une forme gaussienne 139
6.1.3 Formalisme statistique 140
6.1.3.1 Densité de probabilité d'observation 140
6.1.3.2 Estimation des paramètres 141
6.1.3.3 Estimation des paramètres de forme connaissant les paramètres de position 142
6.1.3.4 Estimation des paramètres de position 143
6.1.3.5 Probabilité *a posteriori* des visages 144
6.1.4 Résultats de détection 145
6.1.4.1 Performances sur une base d'images 145
6.1.4.2 Performances sur séquences réelles 146
6.2 Modèle de personne (fusion au niveau objet) 147
6.2.1 Description du modèle 147
6.2.2 Cadre Bayesien 149
6.2.2.1 Définition de la densité de probabilité *a priori* 149
6.2.2.2 Définition de la densité de probabilité d'observation . . 150
6.2.3 Estimation des paramètres 150
6.2.3.1 Échantillonnage par Monte-Carlo 151
6.2.3.2 Calcul rapide de la probabilité d'observation 151
6.2.3.3 Optimisation par réduction du nombre de paramètres . 152
6.2.3.4 Détection des personnes dans l'image 153
6.2.4 Résultats de détection 155

6.3 Suivi de personnes 155
6.3.1 État de l'art du suivi de personnes 157
6.3.2 Suivi par prédiction de mouvement 158
6.3.2.1 Prédiction par régression linéaire sur la position des visages 158
6.3.2.2 Prédiction par historique des trajectoires 159
6.3.2.3 Application au modèle d'ellipse 160
6.3.2.4 Application au modèle de personne 162
6.3.3 Suivi par similarité en apparence 162
6.3.3.1 Signature de couleurs 162
6.3.3.2 Comparaison entre signatures 163
6.3.3.3 Performances de l'association de signatures 164
6.4 Conclusions sur la détection et le suivi de personnes 164

La détection de personnes à l'intérieur du véhicule de transport en commun est un objectif principal de ces travaux. C'est en effet une étape obligatoire pour beaucoup d'applications envisagées, que ce soit dans le domaine de l'aide à l'exploitation avec le comptage de personnes, ou encore dans le domaine de l'aide à la relecture des enregistrements vidéo avec l'indexation des montées de personnes. De manière générale, la détection de personnes est un problème vaste, qui ne possède pas de solution meilleure que les autres pour tous les cas. Chaque situation amène son lot de contraintes et l'apparence des personnes à l'image est spécifique à chaque cas. Les algorithmes doivent donc être pensés de manière adaptée. Pour notre contexte de transport en commun, nous nous limitons à la détection des personnes à la montée dans le véhicule, vues par la caméra chauffeur. Cette caméra est dirigée vers la porte avant du véhicule. Les passagers qui montent dans le véhicule apparaissent alors de face, puis de profil lorsqu'ils disparaissent en dehors du champ. Une personne occupe une proportion très importante de l'image, le cadrage de la caméra étant proche du plan américain (cadrage à hauteur des hanches). Aussi, le nombre de personnes visibles simultanément à l'image peut être important lorsqu'il y a affluence. Ce chapitre présente deux méthodes pour la détection de personnes. Elles sont toutes deux basées sur la combinaison des caractéristiques bas-niveau de teinte chair et de mouvement, mais cette combinaison est réalisée de manière différente. La première méthode fusionne les caractéristiques au niveau de chaque pixel indépendamment, puis les visages sont détectés en estimant les paramètres d'un modèle elliptique de visage. La seconde méthode combine quant-à-elle directement les caractéristiques bas-niveau dans un modèle de personne complet, en estimant les paramètres à partir des deux cartes de probabilités de peau et d'avant-plan.

6.1 Modèle de visage (fusion au niveau pixel)

La première méthode de détection de personne considérée consiste à détecter les visages de personnes en estimant les paramètres d'un modèle d'ellipse de teinte chair. Les

scènes de montée des passagers sont des scènes où la détection des personnes est rendue difficile par leur grand nombre. Cela provoque beaucoup d'occultations entre personnes par rapport à la caméra. Au contraire, les visages des personnes sont bien visibles dans notre contexte de vidéosurveillance, de face ou de profil, et les occultations des visages sont bien moins fréquentes. C'est ce qui a motivé l'étude d'un modèle de visage pour la détection de plusieurs personnes.

Cette méthode de détection de personne se décompose en plusieurs étapes :

1. Les cartes de probabilités de teinte chair et d'avant-plan, dont l'extraction a été présentée dans le chapitre 5, sont fusionnées pour obtenir une carte de probabilités de peau.
2. Une première estimation de la position des visages est ensuite obtenue par un filtrage passe-bas de la carte de probabilités de peau et une recherche des maxima locaux.
3. Pour chaque hypothèse de position de visage, les moments locaux sont estimés par une méthode itérative, pour obtenir les deux axes principaux de l'ensemble des pixels de peau composant le visage.
4. Ces deux axes sont analysés afin de décider si ils représentent une tache de teinte chair qui correspond probablement à un visage.

La méthode proposée fonctionne donc à partir d'une combinaison des critères de couleur des pixels et de forme générale des taches de teinte chair.

6.1.1 Fusion teinte chair et mouvement

La détection de visage envisagée est basée sur la teinte chair, et plus précisément sur le regroupement de zones de teinte chair en objets. Comme évoqué précédemment, la teinte chair est une information efficace pour détecter les pixels des visages. Bien sûr il ne s'agit que d'une information couleur et tous les objets de la scène ayant une couleur similaire sont aussi détectés. Une manière de réduire les fausses détections de pixels de visages est de fusionner l'information de teinte chair avec l'information d'avant-plan. Grâce à cette combinaison, les objets d'arrière-plan de la scène ne sont pas pris en compte dans la détection de pixels de visages.

La fusion elle-même est réalisable simplement car les résultats de la détection de teinte chair et de l'avant-plan sont représentés sous forme de cartes de probabilités. Le résultat de la fusion est représenté lui aussi sous forme d'une carte de probabilités. On effectue pour cela la multiplication des deux cartes pixel à pixel. Il est supposé dans ce cas qu'un pixel de visage est un pixel de teinte chair et un pixel d'avant-plan. La multiplication des probabilités de teinte chair et d'avant-plan correspond alors à la probabilité de peau.

6.1.2 Modèle d'ellipse de peau

Un visage est modélisé par une ellipse de peau, dont on cherche à déterminer la position du centre et les axes. [Ség04] utilise aussi un modèle d'ellipse de teinte chair pour la localisation d'un visage, et détecte une ellipse en passant par la transformée de Hough

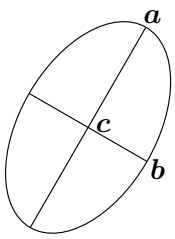


FIG. 6.1 – Modèle d'ellipse de peau

généralisée. [RLM05] propose aussi un algorithme de localisation de visage utilisant la transformée de Hough généralisée pour la détection des ellipses. Cette information de forme est combinée avec deux autres sources d'information que sont l'apparence et la teinte chair, grâce à une représentation sous forme de cartes de probabilités. Un autre modèle similaire a été proposé par [SC00], dans le cas de la détection et du suivi d'un seul visage. La détection et le suivi sont basés sur une mesure des moments d'ordre 1 et 2 du groupe de pixels de teinte chair formant le visage. L'ellipse a l'avantage de pouvoir être paramétrisée simplement, par les moments d'ordres 1 et 2, mesurables directement sur la carte de probabilité de teinte chair. Nous souhaitons utiliser ce même modèle de visage, mais dans un contexte de détection et de suivi de plusieurs personnes simultanément. Dans un tel contexte, nous verrons que la méthode d'estimation des paramètres est très différente d'une simple mesure de moments, car chaque visage doit être localisé de manière robuste aux autres visages présents dans l'image.

Dans l'absolu, seule la localisation des visages nous intéresse, mais l'estimation des axes des ellipses permet d'obtenir la taille et la forme générale des objets de teinte chair, et de discriminer ainsi les objets qui sont des visages des autres objets de la scène. Par exemple, une ellipse dont l'axe horizontal est beaucoup plus grand que l'axe vertical ne correspond certainement pas à une forme de visage. Une information *a priori* sur la forme attendue d'un visage est ici utile pour réduire les risques de fausses détections.

6.1.2.1 Paramétrisation des ellipses

Une ellipse est généralement définie par son centre $\boldsymbol{c} = \begin{bmatrix} c_x \\ c_y \end{bmatrix}$ et ses deux demi-axes $\boldsymbol{a} = \begin{bmatrix} a_x \\ a_y \end{bmatrix}$ et $\boldsymbol{b} = \begin{bmatrix} b_x \\ b_y \end{bmatrix}$, comme représenté figure 6.1. Les coordonnées des demi-axes $\boldsymbol{a}$ et $\boldsymbol{b}$ sont relatives au centre $\boldsymbol{c}$ (afin que $\|\boldsymbol{a}\|$ et $\|\boldsymbol{b}\|$ soient les longueurs des demi-axes).

Nous considérons des ellipses dont les axes ne sont *a priori* pas alignés avec les directions horizontale et verticale (c'est à dire que a_y et b_x peuvent être différents de 0). La représentation de l'ellipse par son centre et ses axes est difficile a manipuler pour l'estimation des paramètres. Une forme équivalente est privilégiée dans ce travail, basée sur la localisation et la répartition des points de l'ellipse.

Le contour de l'ellipse peut être paramétré de la façon suivante :

$$\left\{ \begin{array}{rcl} x & = & c_x + a_x \cos\theta + b_x \sin\theta \\ y & = & c_y + a_y \cos\theta + b_y \sin\theta \end{array} \right. , \quad \theta \in [0; 2\pi[\tag{6.1}$$

A partir de cette paramétrisation du contour, on peut déterminer analytiquement les variances γ_{20}, γ_{02} et la covariance γ_{11} des coordonnées de points de l'ellipse autour du centre $\boldsymbol{c}$ de la manière suivante :

$$\gamma_{ij} = \int_0^{2\pi} \int_0^1 (r(x - c_x))^i (r(y - c_y))^j dr d\theta \tag{6.2}$$

$$\gamma_{ij} = \int_0^{2\pi} \int_0^1 (r(a_x \cos\theta + b_x \sin\theta))^i (r(a_y \cos\theta + b_y \sin\theta))^j dr d\theta \tag{6.3}$$

Le calcul de cette intégrale permet d'exprimer le lien existant entre les demi-axes de l'ellipse et les variances et covariance :

$$\begin{array}{rcl} \gamma_{20} & = & \dfrac{a_x^2 + b_x^2}{3(a_x b_y - a_y b_x)} \\ \gamma_{02} & = & \dfrac{a_y^2 + b_y^2}{3(a_x b_y - a_y b_x)} \\ \gamma_{11} & = & \dfrac{a_x a_y + b_x b_y}{3(a_x b_y - a_y b_x)} \end{array} \tag{6.4}$$

On considère de même les moments de premier ordre μ_x et μ_y, qui sont égaux à c_x et c_y respectivement, car l'ellipse possède une symétrie centrale.

La représentation choisie pour notre modèle d'ellipse contient donc 5 paramètres scalaires, à savoir les moyennes μ_x et μ_y ainsi que les moments d'ordre 2 γ_{20}, γ_{02} et γ_{11}. Pour des considérations de notation, ces paramètres sont regroupés en un vecteur moyen $\boldsymbol{\mu} = \begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}$ et une matrice de covariance $\Gamma = \begin{bmatrix} \gamma_{20} & \gamma_{11} \\ \gamma_{11} & \gamma_{02} \end{bmatrix}$. Cette représentation est bien adaptée à notre problème d'estimation. Ces quantités sont effectivement mesurables à partir de données observées, ce qui n'est pas le cas des paramètres de la représentation sous forme de centre et d'axes.

L'équation 6.4 indique que le passage d'une représentation de l'ellipse à l'autre est réalisable simplement.

6.1.2.2 Analogie avec une forme gaussienne

Le modèle d'ellipse paramétré par les moments d'ordre 1 et 2 peut aussi être interprété comme un modèle de fonction gaussienne de dimension 2, de mêmes paramètres. Bien que les visages observés sont plus proches d'un modèle d'ellipse de peau que d'un modèle de fonction gaussienne bidimensionnelle, cette représentation permet de faciliter les calculs

de preuves, car la fonction gaussienne possède une forme explicite simple et facilement manipulable. En particulier, la fonction gaussienne est continue et s'exprime directement en fonction de ses moments d'ordre 1 et 2. Les deux modèles étant paramétrés par leurs moments aux ordres 1 et 2, il n'y a pas de différence lors de l'estimation des paramètres, qui consistera essentiellement en une mesure robuste de ces moments, comme présenté dans la suite.

La fonction gaussienne $g_{\boldsymbol{v}}$ est associée aux paramètres $\boldsymbol{\mu_v}$ et $\Gamma_{\boldsymbol{v}}$ d'une ellipse $\boldsymbol{v}$. Sa valeur en un pixel $\boldsymbol{k}$ est :

$$g_{\boldsymbol{v}}(\boldsymbol{k}) = \frac{1}{2\pi|\Gamma_{\boldsymbol{v}}|^{1/2}} \exp\left(-\frac{1}{2}(\boldsymbol{k}-\boldsymbol{\mu_v})^T\Gamma_{\boldsymbol{v}}^{-1}(\boldsymbol{k}-\boldsymbol{\mu_v})\right) \tag{6.5}$$

6.1.3 Formalisme statistique

Bien que plusieurs personnes puissent être présentes à l'image simultanément, le modèle d'ellipse ne peut représenter qu'un seul visage. L'estimation des paramètres des ellipses se rapproche donc plus du problème de *clustering* d'un nombre inconnu de *clusters*, que du problème de détermination d'un maximum *a posteriori*.

L'introduction d'un cadre statistique Bayesien permet de formaliser le problème correctement. En dénotant par $\boldsymbol{v}$ un vecteur de paramètres d'une ellipse, et par $\boldsymbol{z}$ la carte de probabilités de peau observée, et en considérant que les ellipses $\boldsymbol{v}$ sont les réalisations d'une variable aléatoire, le théorème de Bayes permet d'écrire :

$$p(\boldsymbol{v}/\boldsymbol{z}) \propto p(\boldsymbol{z}/\boldsymbol{v})p(\boldsymbol{v}) \tag{6.6}$$

La densité de probabilité *a priori* $p(\boldsymbol{v})$ rend compte des positions et formes de visages potentiellement observables, tandis que la densité de probabilité d'observation $p(\boldsymbol{z}/\boldsymbol{v})$ représente la bonne compatibilité entre une hypothèse d'ellipse $\boldsymbol{v}$ et la carte de probabilités de peau $\boldsymbol{z}$. Le produit des deux densités forme la probabilité *a posteriori* $p(\boldsymbol{v}/\boldsymbol{z})$, à la constante $1/p(\boldsymbol{z})$ près.

Dans un premier temps, si l'on considère la détection de toutes les ellipses de peau de l'image, quelque soit leurs formes ou leurs positions, il apparaît que la densité de probabilité *a priori* $p(\boldsymbol{v})$ est constante, et que la probabilité *a posteriori* $p(\boldsymbol{v}/\boldsymbol{z})$ est proportionnelle à la probabilité d'observation $p(\boldsymbol{z}/\boldsymbol{v})$. Cette dernière densité doit être définie.

6.1.3.1 Densité de probabilité d'observation

Comme la scène est susceptible de contenir plusieurs visages, ainsi que d'autres objets de teinte chair, la densité de probabilité d'observation est multimodale. Nous cherchons à définir cette densité de manière à ce que des maxima locaux reconnaissables apparaissent pour des hypothèses $\boldsymbol{v}$ qui correspondent réellement aux ellipses de peau de la scène. Il est montré dans la suite que la corrélation entre la carte de probabilités de peau observée $\boldsymbol{z}$ et la fonction gaussienne $g_{\boldsymbol{v}}$ paramétrée par $\boldsymbol{v}$ est un choix adéquat pour la densité de probabilité d'observation $p(\boldsymbol{z}/\boldsymbol{v})$:

$$\begin{aligned} p(\boldsymbol{z}/\boldsymbol{v}) &\propto \sum_{\boldsymbol{k}\in\mathcal{D}} g_{\boldsymbol{v}}(\boldsymbol{k})\boldsymbol{z}(\boldsymbol{k}) \\ p(\boldsymbol{z}/\boldsymbol{v}) &\propto \sum_{\boldsymbol{k}\in\mathcal{D}} \frac{1}{2\pi|\Gamma_{\boldsymbol{v}}|^{1/2}} exp\left(-\frac{1}{2}(\boldsymbol{k}-\boldsymbol{\mu}_{\boldsymbol{v}})^T\Gamma_{\boldsymbol{v}}^{-1}(\boldsymbol{k}-\boldsymbol{\mu}_{\boldsymbol{v}})\right).\boldsymbol{z}(k) \end{aligned} \tag{6.7}$$

avec $\mathcal{D}$ l'ensemble de définition de l'image, c'est à dire l'ensemble des vecteurs de coordonnées où l'image est définie.

Une manière intuitive de justifier ce choix de densité d'observation provient de la nature de la carte de probabilités de peau. En effet, une hypothèse d'ellipse $\boldsymbol{v}$ peut être interprétée comme un ensemble de pixels de l'image, dont les couleurs suivent une certaine distribution statistique, de densité notée $f_{\boldsymbol{v}}$. Cette distribution peut être approximée par l'histogramme des pixels appartenant à $\boldsymbol{v}$. D'autre part, on note f_P la densité de probabilité de la teinte chair. La probabilité que l'ellipse $\boldsymbol{v}$ soit une ellipse de teinte chair dépend de la similarité entre les densités f_P et $f_{\boldsymbol{v}}$, que l'on peut mesurer par leur corrélation :

$$p(\boldsymbol{z}/\boldsymbol{v}) = \sum_{\boldsymbol{c}\in\mathcal{C}} f_P(\boldsymbol{c}) f_{\boldsymbol{v}}(\boldsymbol{c}) \tag{6.8}$$

avec $\mathcal{C}$ l'ensemble des couleurs. En remplaçant dans l'équation 6.8 $f_{\boldsymbol{v}}(\boldsymbol{c})$ par la valeur de l'histogramme de l'ellipse en $\boldsymbol{c}$, et en remarquant que la carte de probabilités $\boldsymbol{z}$ est construite à partir des valeurs de f_P en chaque point de l'image ($\boldsymbol{z}(\boldsymbol{k}) = f_P(I(\boldsymbol{k}))$), on retrouve l'expression de la corrélation entre l'ellipse et la carte de probabilités de peau :

$$p(\boldsymbol{z}/\boldsymbol{v}) = \frac{1}{Card(E_{\boldsymbol{v}})} \sum_{\boldsymbol{k}\in E_{\boldsymbol{v}}} \boldsymbol{z}(\boldsymbol{k}) \tag{6.9}$$

avec $E_{\boldsymbol{v}}$ l'ensemble des pixels de l'ellipse paramétrée par $\boldsymbol{v}$. Enfin, l'analogie du modèle d'ellipse avec le modèle de fonction gaussienne justifie l'expression de $p(\boldsymbol{z}/\boldsymbol{v})$ définie dans l'équation 6.7.

On admettra intuitivement que $p(\boldsymbol{z}/\boldsymbol{v})$ possède des maxima locaux importants en tout point de l'espace des paramètres correspondant à un objet de teinte chair, car la corrélation entre $\boldsymbol{z}$ et la fonction Gaussienne paramétrée par $\boldsymbol{v}$ y est la plus forte.

Il est supposé dans la suite que $\boldsymbol{z}$ est modélisable par une somme pondérée de N fonctions gaussiennes à deux dimensions de paramètres μ_i et Γ_i (ce qui paraît raisonnable compte tenu de notre modèle de visage) :

$$\boldsymbol{z}(k) = \sum_{i=1}^{N} \alpha_i \frac{1}{2\pi|\Gamma_i|^{1/2}} exp\left(-\frac{1}{2}(\boldsymbol{k}-\boldsymbol{\mu}_i)^T\Gamma_i^{-1}(\boldsymbol{k}-\boldsymbol{\mu}_i)\right) \tag{6.10}$$

6.1.3.2 Estimation des paramètres

Comme plusieurs visages sont susceptibles d'être présents dans l'image, le résultat de l'estimation des paramètres est en réalité un ensemble d'hypothèses de visages pour lesquelles la densité de probabilité *a posteriori* est importante et maximale localement.

L'espace des paramètres des ellipses $\boldsymbol{v}$ est un espace de dimension 5. Une recherche exhaustive des maxima locaux de la densité de probabilité d'observation $p(\boldsymbol{z}/\boldsymbol{v})$ est possible mais très coûteuse en temps de calcul. Une solution alternative consiste séparer l'estimation des 5 paramètres en deux estimations consécutives, d'abord de la moyenne $\boldsymbol{\mu_v}$ puis de la matrice de covariance $\Gamma_{\boldsymbol{v}}$. Cela est rendu possible par deux remarques spécifiques à notre problème :

- La connaissance de la forme des visages, donnée par les estimées de $\Gamma_{\boldsymbol{v}}$ n'est pas absolument nécessaire pour l'estimation des positions $\boldsymbol{\mu_v}$. En réalité, un *a priori* sur la forme des visages suffit à obtenir une première estimation des positions.
- Les visages étant supposés bien séparés les uns des autres, la connaissance des positions $\boldsymbol{\mu_v}$ facilite l'estimation des $\Gamma_{\boldsymbol{v}}$. En effet, pour un visage à une position $\boldsymbol{\mu_v}$ donnée, les paramètres de forme $\Gamma_{\boldsymbol{v}}$ suivent une loi de probabilité quasiment monomodale . C'est à dire que $p(\Gamma_{\boldsymbol{v}}/\boldsymbol{\mu_v})$ a un maximum très reconnaissable pour la valeur réelle de $\Gamma_{\boldsymbol{v}}$.

6.1.3.3 Estimation des paramètres de forme connaissant les paramètres de position

Nous nous plaçons tout d'abord dans le cas où la position $\boldsymbol{\mu_v}$ d'un visage $\boldsymbol{v}$ parmi les visages présents dans l'image a été estimée. A partir de cet information et de la carte de probabilités de peau, les paramètres de formes $\Gamma_{\boldsymbol{v}}$ doivent être estimés.

Connaissant la position $\boldsymbol{\mu_v}$ d'un visage, l'estimation de ses paramètres de formes $\Gamma_{\boldsymbol{v}}$ consiste a maximiser la corrélation entre la fonction gaussienne $g_{\boldsymbol{v}}$ et $\boldsymbol{z}$ en fonction de $\Gamma_{\boldsymbol{v}}$ avec $\boldsymbol{\mu_v}$ fixé.

Si un seul visage était présent dans l'observation $\boldsymbol{z}$, l'estimation de $\Gamma_{\boldsymbol{v}}$ connaissant $\boldsymbol{\mu_v}$ consisterait simplement à calculer la matrice de covariance de $\boldsymbol{z}$:

$$\Gamma_{\boldsymbol{v}} = \sum_{\boldsymbol{k}\in\mathcal{D}} \boldsymbol{z}(\boldsymbol{k})(\boldsymbol{k}-\boldsymbol{\mu_v})(\boldsymbol{k}-\boldsymbol{\mu_v})^T \tag{6.11}$$

L'observation $\boldsymbol{z}$ est ici normalisée, de façon à ce que la somme des $\boldsymbol{z}(\boldsymbol{k})$ sur $\mathcal{D}$ vale 1.

Comme plusieurs visages sont *a priori* présents dans l'image, la matrice de covariance de $\boldsymbol{z}$ ne correspond pas à la matrice de covariance $\Gamma_{\boldsymbol{v}}$ du visage de centre $\boldsymbol{\mu_v}$. L'estimation serait en effet perturbée par les autres visages. La méthode d'estimation proposée dans la suite est une optimisation itérative qui utilise une mesure locale au visage concerné.

Soit W une fenêtre, définie comme une fonction de même domaine de définition que l'observation $\boldsymbol{z}$ et à valeurs réelles, telle que :

$$\sum_{\boldsymbol{k}\in\mathcal{D}} W(\boldsymbol{k}) = 1 \tag{6.12}$$

La matrice de covariance de $\boldsymbol{z}$ locale à la fenêtre W et centrée en $\boldsymbol{\mu_v}$ est définie comme suit :

$$L_{\boldsymbol{z},W} = \sum_{\boldsymbol{k}\in\mathcal{D}} W(\boldsymbol{k})\boldsymbol{z}(\boldsymbol{k})(\boldsymbol{k}-\boldsymbol{\mu_v})(\boldsymbol{k}-\boldsymbol{\mu_v})^T \tag{6.13}$$

Lorsque la fenêtre W est telle que seuls les pixels du visage centré en $\boldsymbol{\mu_v}$ y appartiennent, la contribution des autres pixels de peau est nulle et $L_{\boldsymbol{z},W}$ ne dépend que de $\Gamma_{\boldsymbol{v}}$ et de la forme de la fenêtre. On choisit dans la suite une forme gaussienne pour W. On dénotera par $f_{\mathcal{N}(\boldsymbol{\mu},\Gamma)}$ la fonction gaussienne de centre $\boldsymbol{\mu}$ et de matrice de covariance Γ.

Une suite d'estimées de $\Gamma_{\boldsymbol{v}}$, notées $\widehat{\Gamma_{\boldsymbol{v}(i)}}$, est définie à partir des matrices de covariance locales, utilisant des fenêtres de forme gaussienne.

$$\widehat{\Gamma_{\boldsymbol{v}(0)}} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{6.14}$$

$$W_{(i+1)} = f_{\mathcal{N}(\boldsymbol{\mu_v},\widehat{\Gamma_{\boldsymbol{v}(i)}})} \tag{6.15}$$

$$\widehat{\Gamma_{\boldsymbol{v}(i+1)}} = \alpha.L_{\boldsymbol{z},W_{(i+1)}} \tag{6.16}$$

avec α une constante scalaire ≥ 1 qui permet d'influer sur la vitesse de convergence de la suite. $\widehat{\Gamma_{\boldsymbol{v}(0)}}$ est initialisée à l'identité dans l'équation 6.14, mais le seul critère requis pour cette initialisation est que cette matrice de covariance représente une ellipse de taille plus faible que la taille attendue pour l'ellipse $\boldsymbol{v}$, afin que l'influence des autres ellipses soit négligeable.

L'équation 6.16 se réduit à un calcul de matrice de covariance d'un produit de deux gaussiennes de même centre $\boldsymbol{\mu_v}$ et de matrices de covariance *a priori* différentes. Ce calcul peut se simplifier et la suite peut s'exprimer directement en fonction des matrices de covariance de $\boldsymbol{z}$ et de l'estimation précédente :

$$\widehat{\Gamma_{\boldsymbol{v}(i+1)}} = \frac{\alpha}{2\pi|\widehat{\Gamma_{\boldsymbol{v}(i)}} + \Gamma_{\boldsymbol{v}}|^{1/2}}(\widehat{\Gamma_{\boldsymbol{v}(i)}}^{-1} + \Gamma_{\boldsymbol{v}}^{-1})^{-1} \tag{6.17}$$

(i) étant l'indice de la suite. On admettra la convergence de cette suite vers $\Gamma_{\boldsymbol{v}}$ à un facteur près.

6.1.3.4 Estimation des paramètres de position

L'estimation des paramètres de forme $\Gamma_{\boldsymbol{v}}$ des ellipses de peau suivant l'algorithme qui vient d'être présenté nécessite la connaissance de la position des visages $\boldsymbol{\mu_v}$ pour chaque visage. Ces positions sont estimées à partir de la carte de probabilités de peau observée $\boldsymbol{z}$ et d'information *a priori* sur la forme d'un visage.

Dénotons par Γ_{pr} la matrice de covariance correspondant à la forme d'un visage tel qu'on l'attend *a priori*. Cette matrice est définie manuellement à partir de l'observation des séquences vidéo à traiter, et correspond à la forme *moyenne* des visages observables. La convolution de la carte de probabilités de peau avec une forme gaussienne de covariance Γ_{pr} résulte en une nouvelle carte de probabilités. Celle-ci est interprétée comme la probabilité

en chaque position qu'un centre de visage soit *a priori* présent. La convolution avec la fonction gaussienne de covariance Γ_{pr} revient effectivement à corréler en chaque position l'observation $\boldsymbol{z}$ par un visage *a priori*. Une corrélation forte en une position $\boldsymbol{\mu_v}$ signifie une forte probabilité qu'un visage de forme similaire à Γ_{pr} soit présent en $\boldsymbol{\mu_v}$.

L'estimation des centres à proprement dit consiste en deux étapes.

- Comme plusieurs visages sont susceptibles d'être visibles simultanément, la première étape consiste à sélectionner, à partir de la nouvelle carte de probabilités de centres, un ensemble de centres probables. Ceux-ci sont choisis parmi les maxima locaux de la carte de probabilités de centres. Une manière simple consiste à considérer tous les maxima locaux. La sélection d'un nombre fini des maxima locaux (tout de même supérieur au nombre attendu de visages simultanément présents) dont les probabilités sont les plus fortes permet de réduire considérablement le temps de calcul pour l'estimation des paramètres de forme.
- La seconde étape de l'estimation des centres intervient plus tard, et consiste au contraire à supprimer les hypothèses de centres mal estimées lors de la première étape.

Les maxima locaux de la carte de probabilités de centres sélectionnés comme centres probables sont considérés, et les matrices de covariance correspondantes sont estimés par l'algorithme itératif présenté précédemment. Un ensemble d'estimées de visages est ainsi obtenu.

6.1.3.5 Probabilité *a posteriori* des visages

Jusque là, la seule information *a priori* qui a été prise en compte est celle concernant la forme *a priori* des visages, convoluée à la carte de probabilités de peau pour obtenir la carte de probabilités des centres. Malgré tout, aucune contrainte sur la forme des ellipses de teinte chair à estimer n'a été prise en compte pour l'instant. La carte de probabilités des centres peut effectivement contenir des maxima locaux pour des zones de teinte chair dont la forme est très différente de la forme de visage *a priori*. De plus, une zone de teinte chair de taille plus importante que la taille *a priori* d'un visage sera fortement corrélée à la forme de visage *a priori*. Il est par conséquent nécessaire d'utiliser à nouveau de l'information *a priori* afin d'éliminer les estimées $\boldsymbol{v}$ dont la probabilité d'observation $p(\boldsymbol{z}/\boldsymbol{v})$ est forte (l'algorithme itératif précédent maximisant cette probabilité) mais qui s'avèrent être *a priori* improbables, selon $p(\boldsymbol{v})$. Cette élimination se rapproche d'une application du théorème de Bayes, équation 6.6, avec une densité de probabilité *a priori* uniforme sur l'ensemble des paramètres $\boldsymbol{v}$ possibles pour un visage.

La densité de probabilité *a priori* $\boldsymbol{v} \mapsto p(\boldsymbol{v})$ est définie explicitement, comme une fonction définie sur l'espace de dimension 5 des paramètres, et à valeurs réelles positives. On choisit pour les paramètres de position $\boldsymbol{\mu}_x$ et $\boldsymbol{\mu}_y$ une distribution uniforme sur l'ensemble des points de l'image. La distribution des paramètres de formes γ_{20}, γ_{02} et γ_{11} est une distribution monomodale, par exemple gaussienne trivariée, centrée en Γ_{pr} et de variance définie expérimentalement. Le rôle de $p(\boldsymbol{v})$ étant limité à la suppression des estimées *a priori* improbables, il est suffisant de définir des seuils sur des critères de formes tels que



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

FIG. 6.2 – Détection de visage par le modèle d'ellipse de teinte chair

	Nombre de visages détectés	Taux de détection
Bonnes détections	829	95%
Fausses détections	146	15%

TAB. 6.1 – Performance de la détection de visage avec le modèle d'ellipse sur la base Caltech

le rapport entre la longueur des deux axes et la taille du plus grand axe.

On obtient au finalement un ensemble d'échantillons dont la probabilité *a posteriori* $p(\boldsymbol{v}/\boldsymbol{z})$ est importante, et qui constitue alors notre estimation finale.

6.1.4 Résultats de détection

Le modèle d'ellipse de teinte chair a été validé sur des séquences réelles de transport en commun ainsi que sur une base d'images de test incluant des photos de visages. Une vérité terrain permet de quantifier la performance de l'algorithme de détection.

6.1.4.1 Performances sur une base d'images

La base de données Caltech [FPZ03] contient 873 images de visages, sur un total de 9352 images. La quantification des performances de détection consiste à déterminer le nombre d'images de visages pour lesquelles le visage a été correctement détecté et localisé, ainsi que le nombre d'images autres où un visage a été détecté alors qu'il n'y en avait pas. La figure 6.2 illustre des exemples de bonnes détections des visages (a)-(d), pour des illumination variées et des couleurs de peau différentes, ainsi qu'une fausse détection (e).

Les performances en bonnes détections et fausses détections sont présentées dans le tableau 6.1. Les fausses détections inclut toutes les ellipses détectées qui ne correspondent pas à un visage. En particulier, bien que les photos de la base Caltech sont principalement des visages en gros plan, il arrive que l'algorithme localise mal les visages. Les fausses détections surviennent lorsque l'image inclut des objets dont la couleur est très similaire à la teinte chair, et dont la forme générale est relativement proche de celle d'un visage. L'information couleur n'est donc pas suffisante pour discriminer efficacement les visages des autres objets de teinte chair.



(1) (2)

(3) (4)

FIG. 6.3 – Résultats de détection sur séquences réelles avec le modèle d'ellipse. Les ellipses blanches correspondent aux visages réellement renvoyés par l'algorithme. Les autres ellipses sont des hypothèses avancées puis rejetées par l'algorithme

	Nombre de personnes	Taux de détection
Bonnes détections	333	63.9%
Fausses détections	72	17.8%

TAB. 6.2 – Taux de détection de visages avec le modèle d'ellipse de peau, sur 521 visages

6.1.4.2 Performances sur séquences réelles

Afin de mesurer la performance de l'algorithme de détection sur des séquence réelles, nous avons annoté manuellement la position des visages de personnes pour des séquences de montée dans un véhicule. Cela nous procure une vérité terrain des visages visibles sur les séquences de test. Lorsqu'un visage est caché, par une autre personne par exemple, il n'est pas pris en compte dans les résultats.

Les séquences contiennent au total 521 visages visibles, parmi lesquels 333 ont été correctement détectés et localisés. 72 autres détections ne correspondaient pas à un visage. Cela correspond à un taux de bonnes détections de 63.9%, pour 17.8% de fausses détections. Les résultats sont illustrés par la figure 6.3, et récapitulés dans le tableau 6.2. Les faibles performances ont pour causes différents facteurs. En particulier, le modèle de peau n'arrive pas à intégrer toutes les variations possibles de la teinte chair visibles dans les séquences, malgré l'apprentissage effectué sur la base d'images FERET, à cause des fortes variations en illumination. De plus, la proximité des visages, lorsque le nombre de personnes présentes simultanément est important, entraîne des problèmes de collisions des zones de peau, qui empêche la détection correcte de tous les visages. Enfin, de nombreuses fausses détections apparaissent, principalement à cause des bras nus des passagers. Cela

nous incite à considérer un modèle de personne plus performant, qui modélise une personne par des caractéristiques plus spécifiques, afin d'améliorer les résultats de détection.

6.2 Modèle de personne (fusion au niveau objet)

Comme le montrent les résultats de la détection de personnes basée sur le modèle d'ellipse de peau, la nature du modèle ne permet pas d'éviter certaines fausses détections, en particulier celles des mains de même taille qu'un visage. Il paraît donc nécessaire de remettre en question ce modèle de personne, dont la principale motivation était d'éviter au plus les problèmes d'occultation en se limitant à la description du visage.

Un modèle de personne plus complet que le précédent est introduit ici. Le principe d'estimation des paramètres du modèle reste similaire dans le sens où il est basé sur un cadre Bayesien, confrontant l'information *a priori* à l'information observée. Les différences avec le modèle d'ellipse sont par contre assez nombreuses.

- Premièrement, ce nouveau modèle prend en compte le corps de la personne plus globalement, sans se limiter au visage. Par contre, le visage possède toujours une place à part dans ce modèle car il permet de s'abstraire du problème des occultations.
- Deuxièmement, le modèle se base maintenant sur deux sources d'observation, qui sont la probabilité de peau (comme pour le modèle d'ellipse) et la probabilité d'avant-plan. Ces sources d'information ne sont plus combinées au niveau pixel dans une carte de probabilités résultante, mais directement dans la définition du modèle, au niveau objet.
- Enfin troisièmement, l'estimation des paramètres, bien que basée elle aussi sur un cadre Bayesien, est dans la pratique très différente de celle proposée pour le modèle d'ellipse. La méthode d'estimation itérative proposée pour le modèle d'ellipse est effectivement très spécifique à la nature du modèle. On privilégiera pour ce nouveau modèle complet de personne une méthode d'estimation plus générale, basée sur l'échantillonnage aléatoire de la densité de probabilité *a priori*. Cette méthode d'estimation est adaptée au caractère multimodal de la densité de probabilité *a posteriori*.

6.2.1 Description du modèle

L'apparence des personnes à l'image dans notre contexte de véhicule de transport en commun impose l'utilisation d'un modèle de personne suffisamment général pour prendre en compte la majorité des variations de poses attendues. Les passagers apparaissent à l'image de face puis de profil, à une distance de la caméra proche du plan américain (c'est-à-dire à la hauteur des hanches), comme illustré sur la figure 6.4. Dans un même temps, le modèle doit être assez descriptif pour pouvoir différencier les personnes des autres objets de la scène. La méthode proposée allie ces deux qualités, en combinant plusieurs sources d'observation, pour une bonne discrimination, dans un modèle géométrique relativement simple, pour une bonne généralisation.



FIG. 6.4 – Exemple d'apparence de personnes en plan américain

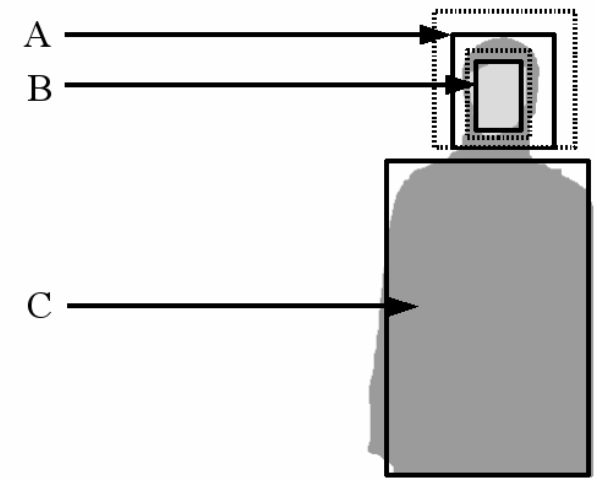


FIG. 6.5 – Modèle de personne. A : région de la tête, B : région du visage, C : région du corps

La figure 6.5 illustre le modèle de personne proposé. Celui-ci est composé de 3 régions rectangulaires, chacune associée à une partie du corps et à une source d'observation. La région A correspond à l'ensemble de la tête, qui doit être un ensemble de pixels d'avant-plan. La région B correspond au visage, et doit être une région de pixels de peau. Enfin la région C correspond au corps, composé de pixels d'avant-plan. Les régions A et B, de la tête et du visage sont encadrées chacune par un autre rectangle de dimensions juste supérieures (en pointillés sur la figure). Les pixels appartenant à ces rectangles encadrants et n'appartenant pas aux rectangles encadrés forment deux régions qui seront notés $\bar{A}$ et $\bar{B}$ respectivement. $\bar{A}$ ne doit pas contenir de pixel d'avant-plan, tandis que $\bar{B}$ ne doit pas contenir de pixel de peau. Ces deux régions seront utilisées lors de l'estimation des paramètres, présentée dans la suite, afin d'éviter d'estimer des régions A et B plus petites que la réalité.

Le modèle de personne est donc paramétré par 12 valeurs scalaires, qui correspondent aux paramètres des 3 rectangles A, B, C. Plus précisément, chaque rectangle du modèle est défini par les coordonnées de son centre, sa hauteur et sa largeur. Le nombre de paramètres du modèle étant relativement faible, il est possible d'utiliser des algorithmes d'estimation classiques en conservant une complexité de calcul assez faible. Nous présentons donc dans la suite notre méthode de localisation de personnes, basée sur un cadre statistique Bayesien et une estimation des paramètres par un échantillonnage de Monte-Carlo.

6.2.2 Cadre Bayesien

Comme pour le modèle d'ellipse de peau, l'estimation des paramètres du modèle humain est basée sur un formalisme Bayesien. On dénote par $\boldsymbol{z}$ l'ensemble des observations à un temps donné, c'est-à-dire les cartes de probabilités de peau et d'avant-plan. $\boldsymbol{v}$ est un vecteur de paramètres pour le modèle de personne. En considérant que ces éléments sont des réalisations de variables aléatoires, on applique le théorème de Bayes :

$$p(\boldsymbol{v}/\boldsymbol{z}) \propto p(\boldsymbol{z}/\boldsymbol{v})p(\boldsymbol{v}) \tag{6.18}$$

La densité de probabilité *a posteriori* $p(\boldsymbol{v}/\boldsymbol{z})$ est proportionnelle au produit de la densité d'observation $p(\boldsymbol{z}/\boldsymbol{v})$ et de la densité *a priori* $p(\boldsymbol{v})$, qui doivent être définies. En particulier, la densité *a priori* fait partie intégrante du modèle, car elle précise les relations spatiales entre les différentes régions du modèle, ainsi que les tailles et positions possibles. Elle précise par exemple de manière explicite que la région de la tête doit se trouver au dessus du corps. La densité de probabilité d'observation $p(\boldsymbol{z}/\boldsymbol{v})$ fait quant-à-elle le lien entre le modèle et les sources d'observation (peau et avant-plan).

6.2.2.1 Définition de la densité de probabilité *a priori*

La densité de probabilité *a priori* est définie de manière empirique, par un ensemble de lois uniformes, en fonction des poses que l'on souhaite tolérer pour notre modèle de personne. Concrètement, le centre de la région du visage suit une loi uniforme sur la moitié supérieure de l'image, tandis que la largeur et la hauteur du visage suivent une loi uniforme

sur une plage de valeurs possibles, avec un ratio largeur/hauteur probable (la hauteur doit être plus grande que la largeur). Les positions de la région de la tête et de la région du corps dépendent de la position du visage, et suivent alors une loi uniforme définie autour de points situés au centre du visage et à une distance fixe sous le visage respectivement.

La densité *a priori* $p(\boldsymbol{v})$ d'un vecteur $\boldsymbol{v}$ issu du modèle n'a que deux valeurs possibles, qui sont soit 0 lorsqu'une des valeurs n'appartient pas au domaine d'une des lois uniformes, soit c, avec c une constante. Une définition plus fine de la densité *a priori* est possible, utilisant des lois non-uniformes, mais le côté empirique de l'approche n'en justifie pas l'intérêt.

6.2.2.2 Définition de la densité de probabilité d'observation

La densité de probabilité d'observation $p(\boldsymbol{z}/\boldsymbol{v})$ est définie en fonction de la bonne concordance entre les régions rectangulaires d'un vecteur de paramètre $\boldsymbol{v}$ et les cartes de probabilités $\boldsymbol{z}$ de peau et d'avant-plan. Comme dans le cas du modèle d'ellipse, la corrélation entre chaque région et la carte de probabilité associée est considérée pour la mesure de probabilité. Notons P_A la corrélation entre la région A correspondant à la tête et la carte de probabilités f_{AP} d'avant-plan associée, c'est à dire la probabilité d'avant-plan moyenne des points de la région A :

$$P_A = \frac{1}{Card(A)} \sum_{\boldsymbol{k} \in A} f_{AP}(\boldsymbol{k}) \tag{6.19}$$

Les probabilités moyennes P_B et P_C sont définies de manière similaire, pour les régions du visage et du corps respectivement (avec la carte de probabilité de peau f_P pour la région du visage). On définit aussi $P_{\bar{A}}$ et $P_{\bar{B}}$, les probabilités moyennes des régions autour du visage et autour de la tête. Par exemple :

$$P_{\bar{B}} = \frac{1}{Card(\bar{B})} \sum_{\boldsymbol{k} \in \bar{B}} f_P(\boldsymbol{k}) \tag{6.20}$$

Le modèle de personne est défini de telle manière que P_A, P_B et P_C soient fortes, tandis que $P_{\bar{A}}$ et $P_{\bar{B}}$ doivent être faibles. On définit alors naturellement la probabilité d'observation d'un vecteur $\boldsymbol{v}$ comme une combinaison de ces probabilités moyennes de chaque région :

$$p(\boldsymbol{z}/\boldsymbol{v}) \propto P_A.P_B.P_C.(1 - P_{\bar{A}}).(1 - P_{\bar{B}}) \tag{6.21}$$

6.2.3 Estimation des paramètres

Maintenant que les densités de probabilités *a priori* $p(\boldsymbol{v})$ et d'observation $p(\boldsymbol{z}/\boldsymbol{v})$ ont été définies pour tout vecteur de paramètres $\boldsymbol{v}$, on peut en déduire, à une constante près, la densité de probabilité *a posteriori* $p(\boldsymbol{v}/\boldsymbol{z})$ comme le produit des deux, grâce au théorème de Bayes (équation 6.18). Deux difficultés interviennent pour l'estimation des paramètres :

– Tout comme dans le cas du modèle d'ellipse de peau, le modèle de personne n'est capable de décrire qu'une seule personne, alors que le contexte impose la présence simultanée d'un nombre inconnu de personnes. Cela se traduit par une densité de probabilité *a posteriori* fortement multimodale. Aussi, l'estimation des paramètres ne se réduit pas à la détermination d'un maximum *a posteriori*, qui correspondrait à la détection d'une seule des personnes visibles. Il faut au contraire considérer chaque mode significatif de la densité, et décider si il s'agit d'une personne ou d'une fausse alarme.
– La dimension du vecteur de paramètres est relativement importante. La recherche des modes significatifs dans un espace de dimension 12 ne peut pas être envisagée de manière exhaustive pour des raisons de temps de calcul. L'échantillonnage par une méthode de type Monte-Carlo est alors envisagé.

6.2.3.1 Échantillonnage par Monte-Carlo

La dimension relativement élevée de l'espace des paramètres incite à envisager un échantillonnage de type Monte-Carlo pour approximer la densité *a posteriori*. Comme on ne peut pas échantillonner directement cette densité, la méthode consiste à d'abord échantillonner la densité *a priori*, puis à pondérer chaque échantillon $\boldsymbol{v}$ par sa probabilité d'observation $p(\boldsymbol{z}/\boldsymbol{v})$.

L'échantillonnage à partir de la variable aléatoire *a priori* est réalisable simplement car la densité *a priori* $p(\boldsymbol{v})$ est composée de lois uniformes. Le tirage d'un échantillon consiste en une suite de tirages aléatoires à partir de lois uniformes, dont les paramètres dépendent généralement d'un tirage précédent. Par exemple, le tirage de la position du corps s'effectue à partir d'une loi uniforme dont les paramètres dépendent du tirage de la position du visage.

L'ensemble $\mathcal{V}$ des échantillons, associés à leur probabilité d'observation, dont le calcul a été détaillé précédemment, forme une approximation discrète de la densité *a posteriori* $p(\boldsymbol{v}/\boldsymbol{z})$.

6.2.3.2 Calcul rapide de la probabilité d'observation

Le modèle de personne étant composé de rectangles dont les côtés sont horizontaux et verticaux, le calcul de la probabilité d'observation peut s'effectuer de manière rapide. Les probabilités moyennes intervenant dans ce calcul sont en effet calculables en un temps constant, quelque soit le rectangle, à condition de précalculer l'*image intégrale* des deux cartes de probabilités. Une méthode d'optimisation similaire a été présentée précédemment pour l'extraction de pixels d'avant-plan (*cf.* 5.4.8).

Considérons l'image intégrale F_P de la carte de probabilités de peau f_P, définie comme suit :

$$F_P(x,y) = \sum_{y'=0}^{y'=y} \sum_{x'=0}^{x'=x} f_P(x',y') \tag{6.22}$$

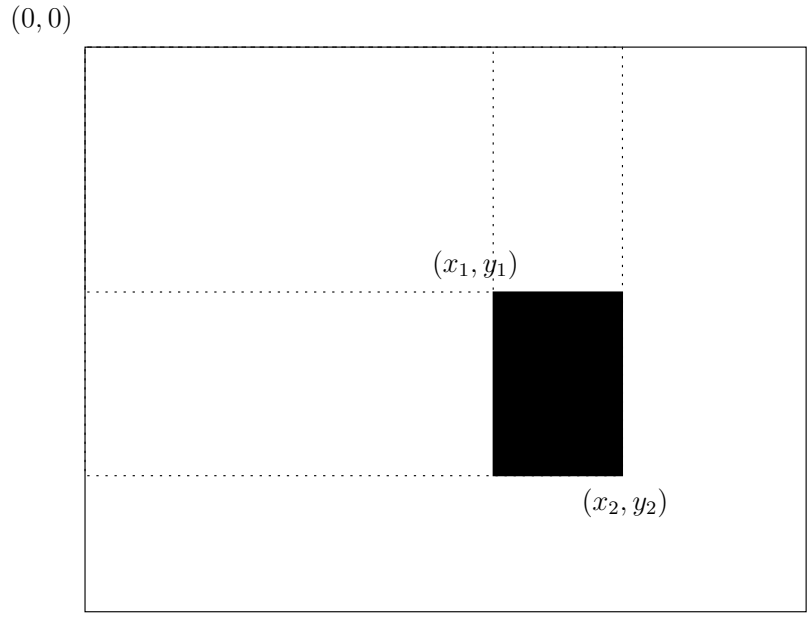


FIG. 6.6 – Calcul rapide de la probabilité moyenne d'un rectangle à partir de l'image intégrale

Le calcul de F_P pour tout les points (x, y) est rapide car il peut s'effectuer de manière récursive. En effet on remarque que pour tout $x > 0$ et $y > 0$,

$$F_P(x, y) = F_P(x-1, y) + F_P(x, y-1) - F_P(x-1, y-1) + f_P(x, y) \tag{6.23}$$

La probabilité moyenne P_R dans un rectangle R quelconque de coordonnées $(x_1, y_1), (x_2, y_2)$ (figure 6.6) est ensuite calculée en très peu d'opérations à partir de F_P :

$$P_R = \frac{F_P(x_2, y_2) - F_P(x_1 - 1, y_2) - F_P(x_2, y_1 - 1) + F_P(x_1 - 1, y_1 - 1)}{(x_2 - x_1 + 1)(y_2 - y_1 + 1)} \tag{6.24}$$

Comme la probabilité d'observation doit être calculée sur un grand nombre d'échantillons, et qu'il s'agit de calculer la probabilité moyenne sur un grand nombre de rectangles sur la même carte de probabilité, cette optimisation s'avère très intéressante. Elle est même indispensable pour que l'algorithme s'exécute en un temps raisonnable.

6.2.3.3 Optimisation par réduction du nombre de paramètres

Le nombre d'échantillons requis pour que $\mathcal{V}$ approxime correctement la densité *a posteriori* est fonction de la dimension du vecteur de paramètres. Il apparaît qu'avec ce modèle de personne de dimension 12, la probabilité qu'un échantillon tiré à partir de la loi *a priori* corresponde aux paramètres d'une personne visible dans l'image est très réduite. Le nombre

important d'échantillons et la taille relativement importante du vecteur de paramètres font que les capacités mémoire de la machine sont rapidement limitées. L'impact sur la vitesse de calcul est aussi important.

Une solution pour réduire le nombre de paramètres, et par la même occasion le nombre d'échantillons nécessaires et ainsi l'occupation mémoire, consiste à ne tirer aléatoirement qu'une partie du vecteur. Dans notre cas, on réduit le tirage aléatoire aux 4 paramètres de la région de la tête. Les autres paramètres du vecteur, qui ne sont pas tirés aléatoirement, sont estimés au moment du calcul de la probabilité d'observation.

Le calcul de la probabilité d'observation est maintenant légèrement différent de celui présenté dans l'équation 6.21. Considérons un vecteur $\boldsymbol{v}$ dont les 4 paramètres de la région de la tête ont été déterminés aléatoirement, mais dont les 8 autres paramètres n'ont pas été fixés. La probabilité d'observation de $\boldsymbol{v}$ doit être la probabilité d'observation maximum en fonction des valeurs possibles pour les 8 paramètres inconnus, connaissant les 4 paramètres de la région de la tête.

La recherche de ce maximum est réalisée en considérant simplement l'ensemble des valeurs possibles (par rapport à la densité *a priori*) pour les 8 paramètres des régions du corps et du visage, et en calculant à chaque fois la probabilité d'observation par l'équation 6.21. On ne considère en réalité qu'un sous ensemble assez restreint des valeurs possibles, car nous nous intéressons plus à la détection des personnes qu'à l'estimation précise des paramètres de chaque personne.

6.2.3.4 Détection des personnes dans l'image

L'ensemble $\mathcal{V}$ des vecteurs de paramètres associés à leurs probabilités d'observation approxime la densité *a posteriori*. Cet ensemble contient un nombre important d'hypothèses de personnes, parmi lesquelles la plupart sont des fausses détections. Aussi, plusieurs échantillons ont généralement des paramètres très proches et représentent la même personne. L'étape de détection des personnes consiste à choisir parmi ces hypothèses quelles sont celles qui représentent effectivement une personne, en évitant les doublons. La densité *a posteriori* est approximée de manière discrète, et contient un bruit important, de sorte que la détection des maxima locaux significatifs n'est pas une méthode satisfaisante pour la détection de personnes. En effet, un mode de la densité, correspondant à une des personnes de l'image, contient *a priori* plusieurs maxima locaux à cause du bruit. De plus, la recherche de maxima dans un espace à grande dimension, sans ordre total et discrétisé de manière irrégulière n'est pas une tâche facile.

La détection des personnes proposée utilise une contrainte supplémentaire sur l'apparence des personnes, qui impose aux visages détectés de ne pas être en collision les uns avec les autres. On fait effectivement l'hypothèse, comme pour le modèle d'ellipse décrit précédemment, que les visages apparaissent bien séparés les uns des autres à l'image. Considérons des échantillons $\boldsymbol{v}_a$ et $\boldsymbol{v}_b$, dont les régions des visages ont pour centre (x_a, y_a) et (x_b, y_b) respectivement, et dont les hauteurs et largeurs sont h_a, l_a et h_b, l_b respectivement. Les régions de visages de $\boldsymbol{v}_a$ et $\boldsymbol{v}_b$ sont en collision lorsque les deux conditions

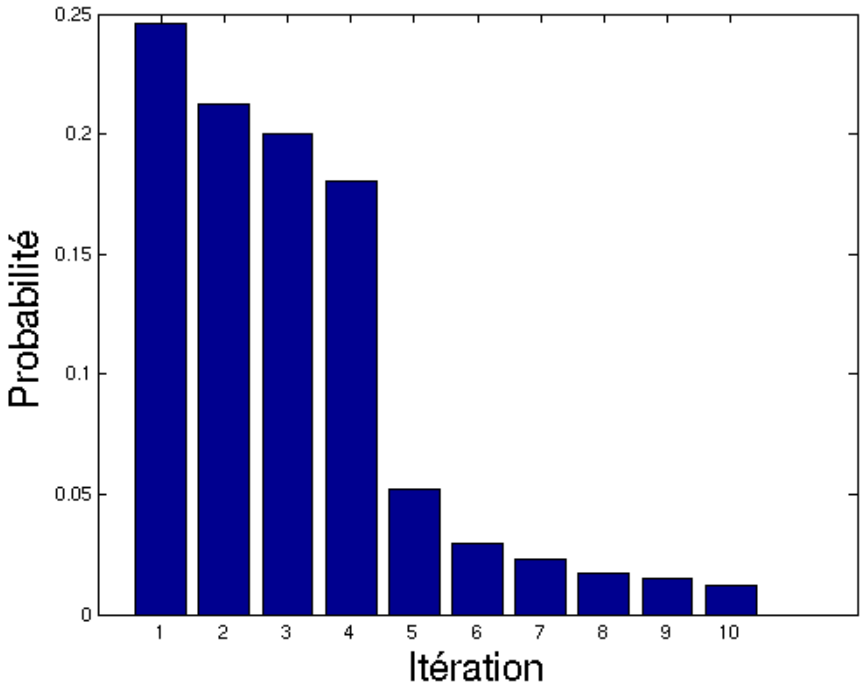


FIG. 6.7 – Probabilités associées aux vecteurs $\boldsymbol{v}_{max}$ pendant l'étape de détection, en fonction de l'itération

suivantes sont vraies :

$$\begin{cases} (x_b - x_a)^2 & < & \left(\dfrac{l_a + l_b}{2}\right)^2 \\ (y_b - y_a)^2 & < & \left(\dfrac{h_a + h_b}{2}\right)^2 \end{cases} \tag{6.25}$$

L'étape de détection de personnes est alors composée de deux étapes répétées itérativement jusqu'à une condition d'arrêt. L'ensemble $\mathcal{P}$, initialisé à l'ensemble vide, contiendra alors les vecteurs des personnes détectées. L'algorithme est le suivant :

1. Choisir dans $\mathcal{V}$ le vecteur $\boldsymbol{v}_{max}$ de probabilité la plus forte qui n'est en collision avec aucun vecteur de $\mathcal{P}$
2. Si la probabilité de $\boldsymbol{v}_{max}$ satisfait la condition d'arrêt, alors la détection est terminée, sinon on ajoute $\boldsymbol{v}_{max}$ à l'ensemble $\mathcal{P}$.

La condition d'arrêt est un seuil sur la probabilité associée à $\boldsymbol{v}_{max}$. La valeur de ce seuil est déterminable facilement, les vecteurs ne correspondant pas à une personne ayant généralement des probabilités au moins 3 fois inférieures à celles des vecteurs correspondant à des personnes.

La figure 6.7 présente un exemple de probabilités obtenues pendant l'étape de détection. Les 4 premières probabilités correspondent effectivement à des personnes visibles à l'image,

	Nombre de personnes détectées	Taux de détection
Bonnes détection	301	97%
Fausses détection	12	4%

TAB. 6.3 – Résultats de détection des personnes à un arrêt avec le modèle de personne complet

tandis que les probabilités obtenues aux itérations suivantes sont celles de vecteurs ne correspondant pas à des personnes. La valeur du seuil pour la condition d'arrêt est choisie à 0.1, ce qui permet la détection des 4 personnes de l'image. Cette valeur de seuil est utilisée pour toutes les séquences.

6.2.4 Résultats de détection

Les performances du modèle de personne présenté sont évaluées dans notre contexte de vidéosurveillance embarquée dans un véhicule de transport en commun. Nous nous intéressons aux séquences de montée des passagers, afin que l'on puisse disposer des cartes de probabilités d'avant-plan pour chaque image. Les séquences vidéo sur lesquelles se base l'évaluation contiennent au total 311 passagers visibles, vus de face ou de profil, plus ou moins proches de la caméra. La figure 6.8 présente des exemples d'images d'une séquence d'arrêt, et le résultat de la détection par le modèle de personne. Seules les régions du visage sont affichées car le modèle simplifié par réduction des paramètres est ici utilisé.

Le tableau 6.3 montre que les résultats obtenus avec la méthode proposée sont très bons.

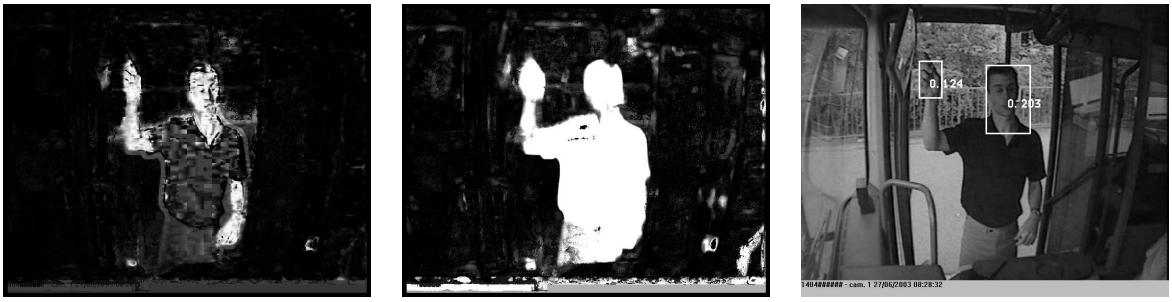
Les fausses détections sont principalement des détections de mains levées, comme on peut le voir sur la figure 6.9. La main est suffisamment séparée du corps et haute dans l'image pour paraître comme un visage. Les régions du modèle de personne correspondant à la tête et au visage ont des probabilités très forte. Dans le calcul de la probabilité totale du modèle, équation 6.21, seule la probabilité de la région du corps P_C est faible, même si elle n'est pas négligeable à cause de l'avant-bras de l'individu qui forme une petite région de pixels d'avant-plan.

6.3 Suivi de personnes

Deux modèles de personne viennent d'être présentés, utilisant tous deux une combinaison des deux sources d'information de bas niveau que sont la teinte chair et l'avant-plan. Ces modèles permettent, en collaboration avec des méthodes d'estimation adaptées, la détection de plusieurs personnes simultanément dans une image. Nous nous intéressons maintenant au suivi de ces personnes au cours d'une séquence vidéo. Le suivi est un problème plus complexe que la détection, car il faut prendre en compte d'autres contraintes fortes de notre contexte. En particulier, les personnes peuvent disparaître du champ de la caméra pendant un instant et réapparaître ensuite lorsqu'elles sont cachées par une autre personne.



FIG. 6.8 – Détection de personnes. (a) : Probabilités de peau. (b) : Probabilités d'avant-plan. (c) : Régions du visage et probabilités associées



(a) (b) (c)

FIG. 6.9 – Fausse détection de main levée. (a) : Probabilités de peau (b) : Probabilité d'avant-plan. (b) : Résultat de la détection de personne

De plus, la fréquence des images dans les séquences vidéo traitées est faible, de l'ordre de 6 images par seconde, ce qui rend difficile la conception d'un algorithme de suivi. Après un bref état de l'art des méthodes de suivi de personnes multiples existant dans la littérature, deux types de suivis sont présentés. Le premier analyse, de manière assez classique, le mouvement d'une personne et prédit la position la plus probable où elle doit apparaître à l'instant suivant. Le second type de suivi analyse l'apparence des personnes détectées et cherche la correspondance entre deux images consécutives.

6.3.1 État de l'art du suivi de personnes

Les méthodes existantes pour le suivi simultané de plusieurs personnes dans une vidéo impliquent généralement que des caractéristiques telles que des histogrammes couleurs, des mesures de corrélation, des estimations de vitesse, ou des mesures de distance entre blobs puissent être utilisées pour suivre chaque personne. Selon les méthodes, ces personnes peuvent être suivies indépendamment les unes des autres, ou de manière unifiée pour une meilleure robustesse aux occultations et aux erreurs d'association.

La majorité des systèmes existants pour le suivi de plusieurs personnes utilisent une représentation des personnes sous forme de blobs calculés par une segmentation des pixels qui diffèrent de l'arrière-plan d'un point de vue statistique [WADP97] [IDB97] [KHM+00]. Une représentation de l'apparence de chaque personne est alors calculée à partir des blobs, suivant des critères de formes ou de couleurs [WADP97]. Cette information sur l'apparence des personnes est alors utilisée pour effectuer les associations au cours du temps. Comme les occultations peuvent perturber l'extraction de l'information sur l'apparence des personnes, d'autres sources d'information peuvent être utilisées pour gagner en robustesse. Ainsi, [KHM+00] considère l'information stéréo couleur pour localiser les blobs, et réalise l'association des personnes entre images de la séquence grâce à l'intersection d'histogrammes en cas d'occultation entre personnes. D'autres sources d'information peuvent aider au suivi de personnes en fonction du contexte. En particulier, le système développé par [IB95] utilise une information *a priori* sur le nombre de personnes présentes dans la scène, qui reste constant tout au long de la séquence pour un environnement clos.

La prédiction de mouvement est une caractéristique qui a été souvent utilisée pour le suivi simultané de plusieurs personnes. Le filtrage de Kalman permet par exemple la prédiction de la position des blobs entre images consécutives [GSRL98]. [HHD00] suit plusieurs personnes entre deux images consécutives dans des scènes en extérieur par des modèles de mouvement au second ordre de blobs correspondant à des parties du corps.

Certaines méthodes de suivi de personnes prennent en compte une durée d'observation plus importante que seulement deux images consécutives, pour un suivi plus robuste. L'algorithme de CONDENSATION [IB98] permet ainsi de conserver plusieurs hypothèses probables de trajectoires pour un objet suivi, et permet par extension, de suivre plusieurs objets aux contours similaires [MB00]. [KI01] montre comment améliorer les systèmes de suivi de plusieurs personnes basés sur des associations inter-images, en ajoutant des contraintes pour les associations sur une fenêtre temporelle de 1 à 5 secondes.

Les très fortes occultations présentes dans notre contexte de transport en commun nous font privilégier un suivi basé sur l'association des objets entre deux images consécutives, sans véritable estimation des trajectoires des personnes tout au long de leur apparition dans la scène. Nous étudions dans ce travail deux méthodes de suivi de personnes, l'une basée sur la prédiction de mouvement, et l'autre basée sur l'apparence en couleur des personnes.

6.3.2 Suivi par prédiction de mouvement

Malgré le faible nombre d'images par seconde de nos séquences vidéo, les passagers montant dans le véhicule suivent un trajet cohérent, qui peut être prédit de manière relativement fiable. Connaissant le trajet d'une personne jusqu'à un temps t de la séquence, le problème consiste à prédire la position de cette personne au temps $t+1$. Le suivi de personnes s'intègre de manière assez naturelle aux deux méthodes de détection de personnes présentées précédemment. Ces deux méthodes sont effectivement basées sur un cadre Bayesien (équations 6.6 ou 6.18), et le suivi par prédiction de mouvement est de manière générale une modification de la densité *a priori* $p(\boldsymbol{v})$. Cette densité *a priori* est modifiée de façon à rendre improbable les positions de l'image qui s'écarte trop d'une trajectoire logique de la personne.

6.3.2.1 Prédiction par régression linéaire sur la position des visages

Étant donné la fréquence des images des séquences à traiter, une prédiction précise de la position d'un visage n'est pas possible. On utilise pour la prédiction un modèle de régression linéaire simple sur la position des visages, qui suppose une trajectoire du visage en ligne droite sur un intervalle de temps de durée r fixé. En dénotant par $\begin{bmatrix} x_t \\ y_t \end{bmatrix}$ la position d'un visage à l'instant t, le modèle cinématique est :

$$y_t = a.x_t + b + e_t \tag{6.26}$$

avec a et b les paramètres à déterminer par les moindres carrés, et e_t un échantillon d'une loi normale centrée, dont on souhaite minimiser la variance.

Les estimateurs des moindres carrés pour a et b sont :

$$\begin{cases} \widehat{a} &= \dfrac{cov(x,y)}{V(x)} \\ \widehat{b} &= -\dfrac{cov(x,y)}{V(x)}\bar{x} + \bar{y} \end{cases} \tag{6.27}$$

avec $V(x)$ la variance des x_t pour $t \in [t-r, t]$, cov(x,y) la covariance des deux coordonnées, et $\bar{x}$ et $\bar{y}$ les coordonnées moyennes.

La durée r doit être la durée maximum pour laquelle il est raisonnable de supposer une trajectoire linéaire du visage. Pour nos séquences, on doit choisir une valeur de r très faible, $r = 3$.

6.3.2.2 Prédiction par historique des trajectoires

L'information sur les autres visages précédemment détectés et suivis au travers de la même caméra permet la prise en compte d'une information *a priori* sur la trajectoire du visage suivi. En effet, dans notre contexte, les passagers montent dans le véhicule en suivant des trajectoires sensiblement comparables. On peut utiliser cette information pour prédire le mouvement d'une personne nouvellement détectée.

L'apprentissage des trajectoires des visages nécessite de disposer d'une séquence d'apprentissage dans laquelle les personnes suivent des trajectoires similaires à la séquence à traiter, et d'une méthode de suivi annexe qui permet de construire les trajectoires. On utilise pour cela la prédiction par régression linéaire présentée précédemment. L'ensemble $\mathcal{T}$, contenant tous les segments de trajectoires estimés, est alors construit. $\mathcal{T}$ contient la position estimée de chaque visage de la séquence à un temps t donné, et sa position estimée au temps suivant $t+1$. On construit ainsi $\mathcal{T}$ comme un ensemble de $N_{\mathcal{T}}$ segments orientés :

$$\mathcal{T} = \{\boldsymbol{a}_i = \begin{bmatrix} x_i \\ y_i \end{bmatrix}, \boldsymbol{a}'_i = \begin{bmatrix} x'_i \\ y'_i \end{bmatrix}, i \in 1 \dots N_{\mathcal{T}}\} \tag{6.28}$$

Le nombre $N_{\mathcal{T}}$ de segments est donc égal au nombre de visages de la séquence d'apprentissage multiplié par le nombre de points de la trajectoire du visage moins un.

Considérons alors un visage de paramètres $\boldsymbol{u}$, à la position $\boldsymbol{a}_{\boldsymbol{u},t} = [x_{\boldsymbol{u},t}, y_{\boldsymbol{u},t}]^T$ au temps t. On prédit sa position au temps $t+1$ à partir de l'ensemble $\mathcal{T}$ comme suit :

$$\boldsymbol{a}_{\boldsymbol{u},t+1} = \frac{\sum_{i=1}^{N_{\mathcal{T}}} k_i.\boldsymbol{a}'_i}{\sum_{i=1}^{N_{\mathcal{T}}} k_i} \tag{6.29}$$

où k_i est un facteur scalaire qui dépend de la distance entre $\boldsymbol{a}_{\boldsymbol{u},t}$ et $\boldsymbol{a}_i$:

$$k_i = \exp(-\|\boldsymbol{a}_{\boldsymbol{u},t} - \boldsymbol{a}_i\|) \tag{6.30}$$

La position prédite $\boldsymbol{a}_{\boldsymbol{u},t+1}$ du visage $\boldsymbol{u}$ au temps $t+1$ est donc une moyenne pondérée des $\boldsymbol{a}'_i$ de l'historique des trajectoires $\mathcal{T}$.

6.3.2.3 Application au modèle d'ellipse

La prédiction de la position des visages s'intègre de manière assez naturelle à l'algorithme de détection de visages utilisant le modèle d'ellipse. En effet, l'algorithme itératif permettant de calculer la matrice de covariance locale des probabilités de peau (équations 6.14, 6.15, 6.16) peut être légèrement modifié pour prendre en compte une initialisation et une estimation de la position, en plus des paramètres de forme. On peut effectivement mesurer, tout comme pour la covariance, les moments du premier ordre locaux à une fenêtre W sur la carte de probabilités de peau observée $\boldsymbol{z}$:

$$\boldsymbol{\mu}_{\boldsymbol{z},W} = \sum_{\boldsymbol{k}\in\mathcal{D}} W(\boldsymbol{z})\boldsymbol{z}(\boldsymbol{k})\boldsymbol{k} \tag{6.31}$$

Notons $\boldsymbol{\mu}_{pr}$ la position prédite d'un visage $\boldsymbol{v}$ et Γ_{pr} la matrice de covariance prédite, égale à la matrice de covariance estimée au temps précédent. Les équations suivantes forment une suite en (i) permettant d'estimer les paramètres de position $\boldsymbol{\mu}_{\boldsymbol{v}}$ et de forme $\Gamma_{\boldsymbol{v}}$:

$$\begin{array}{rl} \text{Initialisation} & \left\{ \begin{array}{rcl} \widehat{\boldsymbol{\mu}_{\boldsymbol{v}}}_{(0)} & = & \boldsymbol{\mu}_{pr} \\ \widehat{\Gamma_{\boldsymbol{v}}}_{(0)} & = & \Gamma_{pr} \end{array} \right. \\ \text{Mise à jour} & \left\{ \begin{array}{rcl} W_{(i+1)} & = & f_{\mathcal{N}(\widehat{\boldsymbol{\mu}_{\boldsymbol{v}}}_{(i)},\widehat{\Gamma_{\boldsymbol{v}}}_{(i)})} \\ \widehat{\boldsymbol{\mu}_{\boldsymbol{v}}}_{(i+1)} & = & \alpha.\boldsymbol{\mu}_{\boldsymbol{z},W_{(i+1)}} \\ \widehat{\Gamma_{\boldsymbol{v}}}_{(i+1)} & = & \alpha.L_{\boldsymbol{z},W_{(i+1)}} \end{array} \right. \end{array} \tag{6.32}$$

Lorsque les prédictions $\boldsymbol{\mu}_{pr}$ et Γ_{pr} sont assez proches de leurs valeurs réelles $\boldsymbol{\mu}_{\boldsymbol{v}}$ et $\Gamma_{\boldsymbol{v}}$, la tache de peau observée à travers $\boldsymbol{z}$ et la fenêtre W s'intersectent, ce qui permet la convergence de la suite. Il arrive que la prédiction soit mauvaise (lors d'un changement brusque de direction du visage suivi par exemple) auquel cas l'algorithme peut converger vers une tache de peau de probabilité très faible, ou *a priori* peu probable de représenter un visage. Il est alors nécessaire de considérer la probabilité des paramètres estimés $\widehat{\boldsymbol{v}}$ afin de déterminer le succès ou non de l'estimation. Lorsque la probabilité d'observation $p(\boldsymbol{z}/\widehat{\boldsymbol{v}})$ (équation 6.7) est en dessous d'un seuil ou que les dimensions de l'ellipse estimée sont *a priori* improbables, on considère que l'estimation a échoué, et que le visage suivi a disparu.

La figure 6.10 illustre des résultats obtenus pour le suivi de personnes par prédiction de mouvement, en utilisant le modèle d'ellipse pour l'étape de détection. Dans la séquence considérée, la taille minimale des visages pour la détection a été choisie de façon à ne détecter que les personnes très proches de la caméra. De ce fait, le détecteur de visages n'est pas perturbé par les visages présents en arrière-plan, et la séquence ne comporte alors pas de croisements entre visages détectés. Le suivi de visage fonctionne correctement dans ce cas. Les quatre personnes de la séquence sont effectivement suivies lors de leur montée dans le véhicule. Pour des cas plus complexes comportant des croisements entre les visages détectés, les pertes de cibles sont nombreuses. Nous verrons dans le chapitre suivant que

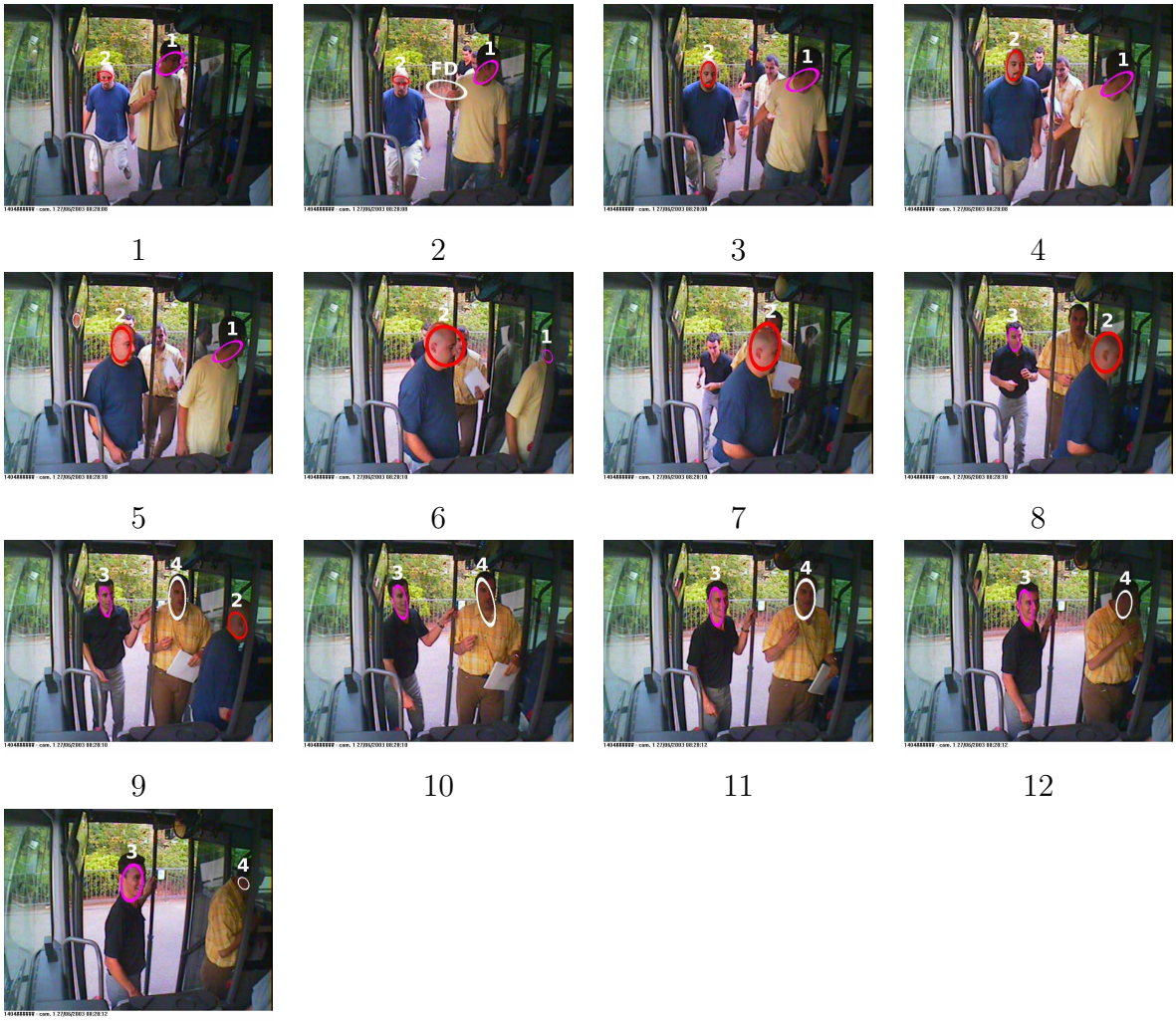


FIG. 6.10 – Suivi de visages par prédiction de mouvement

l'application de comptage de personnes que nous envisageons se contente d'un suivi entre images consécutives, dans une zone de l'image particulière où les croisements sont moins fréquents. Ces pertes de suivi ont donc peu de conséquences pour le comptage de personnes.

6.3.2.4 Application au modèle de personne

La prédiction de la position des visages peut aussi s'intégrer facilement à la méthode de détection de personnes basée sur le modèle complet de personne. Il s'agit dans ce cas de modifier directement, pour chaque personne détectée au temps t, la densité de probabilité *a priori* $\boldsymbol{v} \mapsto p(\boldsymbol{v})$ pour le temps $t+1$. Cette densité est modifiée de telle façon à ce que seules des positions proches de la position prédite soit tirées aléatoirement. Si une personne est détectée parmi ces échantillons de la loi *a priori*, on considère qu'il s'agit de la même personne qu'au temps $t+1$. Néanmoins, ce type de suivi n'a pas été implémenté pour le modèle de personne complet. La nature du modèle et les contraintes des séquences vidéo nous font privilégier un suivi par similarité en apparence, comme présenté dans la suite.

6.3.3 Suivi par similarité en apparence

Le faible nombre d'images par seconde dans les séquences traitées est la cause principale des pertes de suivi que l'on peut rencontrer. Cela nous incite à considérer une méthode de suivi qui n'est pas basée sur la prédiction du mouvement, mais sur l'*apparence* des personnes et leur association entre deux images. L'idée consiste à extraire de l'image des caractéristiques reflétant l'apparence des personnes, suffisamment robustes pour que l'association d'une image à l'autre soit possible malgré les variations de posture. Ces caractéristiques doivent aussi être suffisamment discriminantes entre les personnes afin que l'association fonctionne.

6.3.3.1 Signature de couleurs

Afin d'être robuste aux variations de poses que peut adopter une personne montant dans le véhicule, les caractéristiques que l'on considère pour décrire l'apparence sont basées sur les couleurs des vêtements de la personne. Nous construisons un vecteur contenant les couleurs les plus représentatives de la personne, que nous appelons signature de couleurs.

La signature est calculée à partir de la région du corps du modèle de personne présenté précédemment. Les pixels de cette région incluent non seulement les pixels des vêtements de la personne, mais aussi des pixels de peau, des pixels du fond de la scène et souvent des pixels appartenant aux autres personnes de la scène, lorsqu'il y a occultation entre personnes. Seules les couleurs des vêtements de la personne considérée nous intéressent pour le calcul de la signature. Par conséquent, les pixels de peau et d'avant-plan sont tout d'abord éliminés grâce aux cartes de probabilités de peau et d'avant-plan.

Les pixels qui n'ont pas été éliminés sont quantifiés en un nombre fixe M de couleurs, par l'algorithme k-means. On obtient alors un vecteur S_n de M couleurs pour chaque personne n détectée :

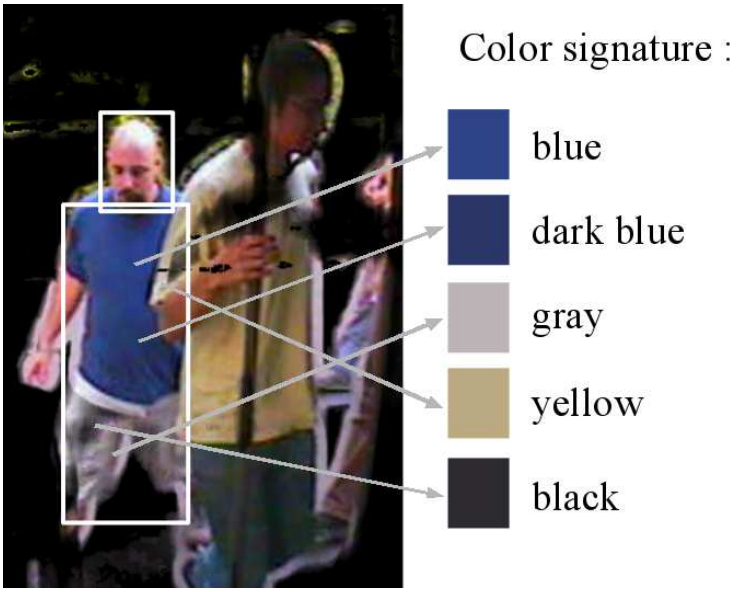


FIG. 6.11 – Signature de couleurs obtenue par segmentation k-means

$$S_n = \{\boldsymbol{c}_1^n, \boldsymbol{c}_2^n, \ldots, \boldsymbol{c}_M^n\} \tag{6.33}$$

La signature S_n obtenue contient les couleurs les plus représentatives de la région du corps. Comme illustré sur la figure 6.11, le vecteur S_n peut contenir aussi des couleurs qui sont issues des vêtements d'une autre personne. Ces couleurs ne peuvent pas être éliminées par les cartes de probabilités de peau ou d'avant-plan. Elles font partie de la signature de la personne, mais la comparaison des signatures entre deux images doit être robuste à ces couleurs supplémentaires. Cette remarque nous incite aussi à choisir un nombre M de couleurs assez important pour les signatures, supérieur au nombre de couleurs représentatives attendu sur le corps d'une personne. En effet, si M est trop petit, l'algorithme k-means risque de sélectionner des couleurs qui font partie d'une autre personne que la personne n considérer, et d'ignorer certaines couleurs de n. On choisit $M = 5$.

6.3.3.2 Comparaison entre signatures

La comparaison des signatures entre deux images de la séquence vidéo permet d'associer les personnes détectées dans une image aux personnes détectées dans l'autre image, et constitue ainsi un algorithme de base pour réaliser le suivi. On définit pour cela une mesure de distance entre signatures de couleurs, qui doit être robuste aux couleurs parasites issues des objets autres que la personne considérée.

Tout d'abord, la distance entre une couleur $\boldsymbol{c}$ et une signature S_n est définie comme le minimum des distances euclidiennes entre $\boldsymbol{c}$ et les $\boldsymbol{c}_i^n$, dans l'espace de chrominance RG-normalisé :

$$d(\boldsymbol{c}, S_n) = min(\|\boldsymbol{c} - \boldsymbol{c}_i^n\|, i = 1 \dots M) \tag{6.34}$$

On suppose que deux signatures peuvent être associées entre elles lorsqu'une certaine proportion des couleurs de chaque signature sont similaires. Pour $M = 5$, on décide que 3 couleurs similaires suffisent pour que deux signatures soient similaires. La distance $D(S_n, S_m)$ entre deux signatures S_n et S_m est alors donnée par la somme des 3 distances minimales $d(\boldsymbol{c}_i^n, S_m)$.

6.3.3.3 Performances de l'association de signatures

Afin de quantifier les performances des signatures de couleurs et de la distance entre signatures, les personnes détectées dans chaque paire d'images consécutives sont associées les unes avec les autres. Lorsque la distance $D(S_n, S_m)$ dépasse un seuil donné, aucune association n'est effectuée, ce qui signifie qu'une personne a disparu ou apparu. Seules les images où au moins deux passagers sont visibles simultanément sont considérées dans cette expérience. Des 129 associations possibles, 112 (87%) ont été détectées correctement.

6.4 Conclusions sur la détection et le suivi de personnes

Ce chapitre a présenté deux méthodes de détection de plusieurs personnes, adaptées à notre contexte de véhicule de transport en commun.

La première méthode est basée sur la modélisation du visage par une ellipse de peau. La détection de telles ellipses dans l'image implique tout d'abord la localisation des pixels de peau à partir des cartes de probabilités de teinte chair et d'avant-plan, dont l'extraction a été présentée dans le chapitre 5. Grâce à l'algorithme itératif proposé, calculant les axes des ellipses à partir de matrices de covariance locales à la région du visage, la localisation des visages et leur forme générale peut être estimée. La simplicité du modèle rend difficile la discrimination entre les visages et d'autres objets de teinte chair de l'image tels que les mains.

Ce défaut de premier modèle proposé nous a incité à définir un modèle de personne plus complet, pour lequel des contraintes géométriques entre les deux sources d'information sont définies explicitement. Pour ce second modèle, la combinaison des pixels de peau et d'avant-plan n'est plus effectuée avant l'étape de détection, mais directement pendant l'estimation des paramètres, par l'intermédiaire de la probabilité d'observation définie. L'estimation des paramètres est quant-à-elle réalisée par un échantillonnage de Monte-Carlo suivi d'une sélection des échantillons les plus probables pour lesquelles les visages ne sont pas en collision. Les performances de ce second modèle sont supérieures au premier, car les risques de fausse détection sont réduits. Alors qu'avec le modèle de visage, les performances étaient de 63.9% de bonnes détections et 17.8% de fausses détections, la seconde méthode permet d'atteindre un taux de bonnes détections de 97% pour 4% de mauvaises détections, comme récapitulé dans le tableau 6.4.

Modèle	Taux de bonnes détections	Taux de fausses détection
Ellipse de peau	63.9%	17.8%
Personne	97%	4%

TAB. 6.4 – Comparaison des performances en détection de personnes entre le modèle d'ellipse de peau et le modèle de personne complet

Aussi, certains résultats préliminaires ont été présentés pour le suivi de personnes, bien que dans ce travail l'accent ait été mis beaucoup plus sur l'étape de détection que sur l'étape de suivi. Le problème du suivi de plusieurs visages simultanément a été abordé au travers de deux approches. La première est basée sur la prédiction du mouvement des visages, soit par un modèle cinématique supposant une trajectoire rectiligne des visages sur un temps court, soit par un apprentissage des trajectoires passées. La seconde méthode se sert du modèle complet de personne pour extraire une signature de couleur sur chaque personne détectée. L'association entre les personnes détectées dans deux images à des temps différents est alors réalisée grâce à une mesure de similarité entre signatures. La correspondance entre personnes ainsi réalisée entre images consécutives va constituer la partie principale de l'application de comptage de personnes.

Chapitre 7

Applications

Sommaire

7.1 Compression sélective 168
7.1.1 Principe 168
7.1.2 Extraction des zones d'intérêt 170
7.1.3 Les différents préfiltrages 171
7.1.3.1 Le filtre DCT 171
7.1.3.2 Le filtre gaussien 173
7.1.3.3 Comparaison entre les deux préfiltrages en compression d'image 175
7.1.3.4 Comparaison entre les deux préfiltrages en compression vidéo 176
7.1.3.5 Influence de la variance du préfiltrage gaussien sur la taille de la vidéo compressée 177
7.1.4 Conclusion sur la compression sélective 179
7.2 Comptage de personnes 179
7.2.1 Passage d'un segment 180
7.2.2 Performances du comptage 181
7.2.3 Conclusion sur l'application de comptage de personnes 181
7.3 Conclusions sur les applications développées 181

L'analyse des séquences vidéo de transport en commun nous a permis de définir des outils pour l'extraction d'objets d'intérêt dans la scène. A partir d'information bas-niveau et d'une modélisation des objets, leur détection et leur localisation est rendue possible. Les travaux ont porté principalement sur la détection des personnes dans les séquences d'arrêt. Les algorithmes présentés dans le chapitre 5, permettant d'extraire l'information bas-niveau, ont aussi leur importance à part entière, et leur résultat peut être directement exploité pour construire des applications. Dans ce chapitre, les principales applications développées pour le système de surveillance sont présentées. Elles sont basées sur l'extraction des objets et des caractéristiques bas-niveau que nous avons présentée précédemment.

Les deux applications sur lesquelles nous nous focalisons ici répondent à deux besoins bien distincts.

1. La première application répond au besoin technique de réduire au maximum la place occupée par les images. Le système de surveillance possède une fonction qui lui permet de transmettre en temps-réel les images provenant des caméras à un poste fixe, central au parc de véhicules. Cela permet à un opérateur de visionner les images de surveillance de n'importe quel véhicule en temps réel lorsque c'est nécessaire. La transmission est bien entendu réalisée via une liaison sans fil qui, pour des raisons de technologie et de coût, possède un débit très faible. Les images doivent donc être fortement compressées pour que la fréquence des images transmises soit raisonnable. Une compression sélective des images, permettant de réduire la distortion dans les zones d'intérêt telles que les visages est présentée.
2. La seconde application a pour but de réaliser des statistiques sur le fonctionnement d'un réseau de transport en commun. Nous souhaitons réaliser une application capable d'estimer automatiquement le taux d'occupation du véhicule à chaque instant. Du point de vue de l'exploitation du réseau, les statistiques permettent de réguler le nombre de véhicules nécessaire sur chaque ligne, et d'optimiser ainsi les horaires de passage. La détection des personnes pendant les arrêts du véhicule est donc très utile ici.

7.1 Compression sélective

Les périphériques de stockage standard actuellement disponibles sur le marché ont une capacité suffisamment grande et un coût relativement faible pour qu'il ne soit pas nécessaire de trop compresser les images de surveillance pour les stocker. La compression vidéo utilisée est de type MJPEG, où chaque image est compressée indépendamment, sans tenir compte des redondances temporelles propres à la vidéo. Même avec un taux de compression assez faible, la capacité de stockage des images atteint aisément 48 heures pour 4 caméras, avec une cadence de 16 images par seconde par caméra.

La transmission en temps réel des images est un tout autre problème, car elle doit être réalisée grâce à une liaison sans fil bas-débit de type GSM ou GPRS. La compression des images est ici un problème plus délicat. Les algorithmes de compression vidéo de type MPEG, prenant en compte la redondance temporelle sont une première solution, mais nous souhaitons explorer les améliorations possibles pour augmenter la cadence des images obtenues tout en conservant une qualité visuelle acceptable pour une bonne exploitation.

7.1.1 Principe

Le principe de la compression sélective est de réaliser une compression d'image qui permette de contrôler la distortion en fonction de zones d'intérêt. Pour une application de vidéosurveillance, on souhaite typiquement une bonne visibilité des visages, tandis que certains éléments de la scène, comme l'intérieur du véhicule, ont moins d'importance et

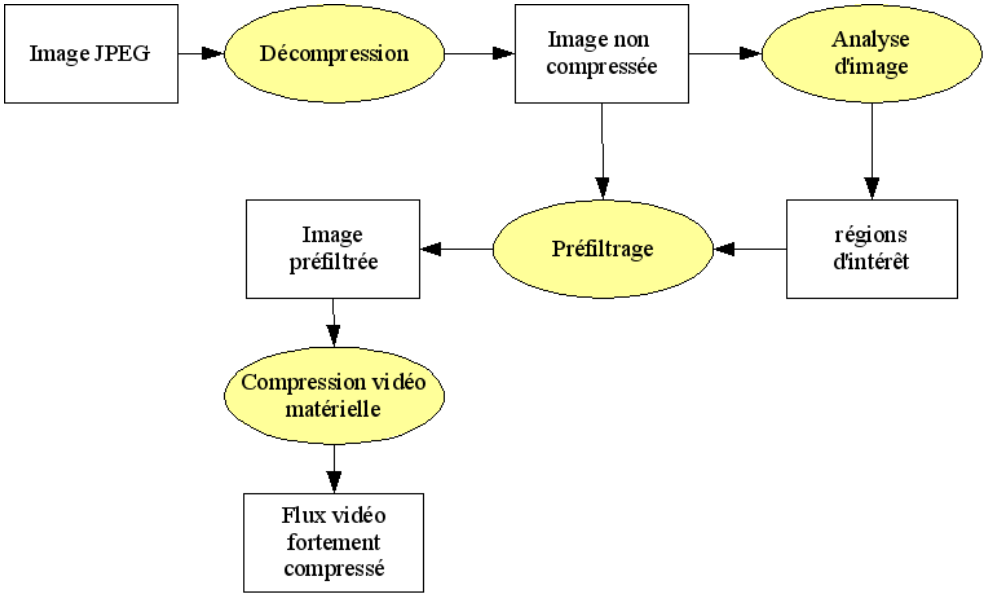


FIG. 7.1 – Schéma de la compression sélective par préfiltrage des images

peuvent être dégradés en qualité. Les algorithmes d'extraction de caractéristiques bas-niveau, que nous avons présentés dans le chapitre 5 vont permettre de définir automatiquement ces zones d'intérêt.

La spécification de zones d'intérêt pour la compression vidéo n'est pas un problème nouveau. La norme MPEG-4 inclut d'ailleurs la définition d'objets vidéo, pour lesquels les paramètres de compression peuvent différer du reste de la scène. L'inconvénient majeur des objets MPEG-4 est qu'il est nécessaire de transmettre le masque de l'objet en même temps que le flux compressé, et la taille de ce masque n'est pas négligeable lorsque l'on souhaite obtenir une vidéo compressée pour une transmission à très bas débit. Les expériences menées au laboratoire [Cot05] ont montré que la compression MPEG-4 avec objets ne donnait pas un gain de compression suffisant pour des objets de formes quelconques.

La compression des images à proprement parler doit être suffisamment rapide pour ne pas surcharger le processeur. Une solution performante consiste à utiliser une compression matérielle standard de type MPEG, mais de réaliser un préfiltrage des images de façon à dégrader intentionnellement les régions qui sont en dehors des zones d'intérêt.

La figure 7.1 présente le principe général de la compression sélective par préfiltrage des images. L'étape d'analyse des images permet d'extraire les zones d'intérêt pour lesquelles la dégradation de l'image doit être minime. Deux types de filtrage sont évalués dans la suite.

7.1.2 Extraction des zones d'intérêt

Parmi les éléments de la scène extraits par analyse d'images, on s'intéresse particulièrement aux visages des personnes et à la partie vitrée du véhicule pour former les différentes zones d'intérêt dans l'image. Les visages sont les éléments de la scène où la distortion doit être minime après compression. Le paysage extérieur est moins intéressant sémantiquement, mais on souhaite tout de même conserver une qualité correcte dans les régions correspondantes de l'image, afin qu'un opérateur puisse être capable de localiser le véhicule. Enfin l'intérieur du véhicule est la région de l'image qui présente le moins d'intérêt pour exploiter les images, et une forte dégradation de la qualité d'image dans cette zone est tolérée. Les pixels de chaque image sont donc séparés en trois classes, dont l'importance est plus ou moins forte d'un point de vue exploitation des images. Par ordre décroissant d'importance, ces classes sont :

1. Les visages
2. Le paysage extérieur
3. L'intérieur du véhicule

L'extraction de ces trois classes est réalisée à partir des algorithmes présentés dans le chapitre 5. En ce qui concerne la classe des visages, on se limite à la seule détection de peau. En effet la détection de personnes qui a été développée dans le chapitre 6 n'est adaptée qu'aux arrêts du véhicule, tandis que la compression sélective doit fonctionner en toutes circonstances. La détection de teinte chair est aussi plus rapide que la détection de visages. L'inconvénient majeur de ce choix est que nous incluons alors dans la première classe toutes les zones de teinte chair de l'image, telles que les mains ou les jambes nues. Cela aura pour seule conséquence de conserver une qualité d'image forte pour des zones de l'image qui n'ont pas forcément un intérêt très important.

La classe du paysage extérieur est extraite par l'algorithme de détection de vitre, qui a été présenté dans la section 5.2 du chapitre 5.

Enfin, la troisième classe, correspondant à l'intérieur du véhicule, regroupe tous les pixels qui n'ont pas été attribués à l'une des deux premières classes.

Un filtrage morphologique est réalisé sur la classe des visages afin de corriger les imperfections du détecteur basé uniquement sur la teinte chair, qui ne détecte pas certaines parties du visage comme les yeux, et qui produit une carte de probabilité de teinte chair assez bruitée. On réalise donc une fermeture morphologique de la carte de probabilité de teinte chair.

La figure 7.2 illustre les cartes de probabilités utilisées pour associer chaque pixel à une classe. Les cartes sont seuillées, puis on associe chaque pixel à la classe correspondante. S'il y a une ambiguïté entre deux classes, on associe le pixel à celle pour qui on dégrade le moins l'information. Par exemple, lorsqu'un visage est devant une vitre, on associe les pixels du visage à la classe des visages, afin que la qualité de l'image compressée soit maximale au niveau du visage. ni de teinte



(a) (b) (c)

FIG. 7.2 – Exemple d'image et cartes de probabilités associées pour la compression sélective. (a) : image originale. (b) : teinte chair. (c) : extérieur

7.1.3 Les différents préfiltrages

Le préfiltrage des images doit permettre une meilleure compression de la vidéo par un algorithme standard de type MPEG, tout en conservant une qualité acceptable. Un compromis doit donc être trouvé entre le résultat visuel du filtrage et le taux de compression obtenu. Un paramètre doit permettre de dégrader l'image plus ou moins en fonction de la classe de chaque pixel. Les filtres que nous considérons sont le lissage gaussien et le filtre DCT.

7.1.3.1 Le filtre DCT

La transformée en cosinus discrète (DCT) [NA74] est utilisée pour la compression d'images par des algorithmes standard tels que JPEG. La quantification des coefficients de la DCT est l'étape la plus importante, avec le sous-échantillonnage de la chrominance, dans l'algorithme JPEG, permettant d'éliminer de l'information pour réduire la taille des données.

En réalisant un préfiltrage de l'image utilisant une DCT, on peut dégrader l'image dans les régions peu intéressantes de manière similaire à l'algorithme JPEG, mais avec des paramètres différents suivant la classe des pixels. La plupart des algorithmes classiques de compression vidéo tels que MPEG utilisent aussi une quantification des coefficients DCT sur les images clés, ce qui laisse présager un bon taux de compression au moins pour ces images clés. En reproduisant les étapes de DCT et de quantification propres aux algorithmes de compression standard, on réalise le préfiltrage le mieux adapté à notre problème, de même que si l'on modifiait directement l'algorithme de compression pour autoriser des paramètres de quantification différents pour chaque bloc de pixels. L'inconvénient est bien entendu le temps de calcul de la DCT, qui est long par rapport à d'autres filtres plus simples comme le moyenneur ou le médian. Le préfiltrage par DCT nous permet surtout de savoir quel type de résultat est possible avec cette technique, en terme de qualité d'image et taux de compression, et permet la comparaison avec les préfiltrages plus simples.

Nous utilisons, comme pour l'algorithme JPEG, un découpage de l'image en blocs de

8×8 pixels pour chaque composante couleur dans l'espace YUV. La DCT bidimensionnelle de chaque bloc B est alors calculée :

$$DCT^B(i,j) = \frac{1}{\sqrt{2.N_B}} C(i)C(j) \sum_{x=0}^{N_B-1} \sum_{y=0}^{N_B-1} B(x,y).\cos\left(\frac{(2x+1)i\pi}{2N_B}\right)\cos\left(\frac{(2y+1)j\pi}{2N_B}\right) \tag{7.1}$$

avec $N_B = 8$, la largeur (et la hauteur) d'un bloc et C la constante suivante :

$$C(x) = \begin{cases} \frac{1}{\sqrt{2}} & \text{pour} \quad x = 0 \\ 1 & \text{pour} \quad x > 0 \end{cases} \tag{7.2}$$

On dégrade ensuite l'image en quantifiant les coefficients la DCT. Pour cela, on utilise une table de quantification Q_B de la même taille que le bloc. Les coefficients quantifiés DCT_Q^B sont obtenus par la formule suivante :

$$DCT_Q^B(i,j) = \text{round}\left(\frac{DCT^B(i,j)}{Q_B(i,j)}\right).Q_B(i,j) \tag{7.3}$$

avec round la fonction renvoyant l'entier le plus proche.

La table de quantification Q_B dépend de la dégradation que l'on souhaite effectuer sur le bloc B. Elle est construite à partir d'une table standard prédéfinie Q , et d'un facteur d'échelle s_B, qui va dépendre de la classe des pixels (visage, intérieur, extérieur) de B :

$$Q_B(i,j) = Q(i,j).s_B \tag{7.4}$$

La table de quantification est différente pour le plan de luminance Y et les plans de chrominance U et V. La table standard pour la luminance est :

$$\begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix} \tag{7.5}$$

tandis que pour la chrominance, la table standard est :

$$\begin{bmatrix} 17 & 18 & 24 & 47 & 99 & 99 & 99 & 99 \\ 18 & 21 & 26 & 66 & 99 & 99 & 99 & 99 \\ 24 & 26 & 56 & 99 & 99 & 99 & 99 & 99 \\ 47 & 66 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \end{bmatrix} \tag{7.6}$$

Quant-au facteur d'échelle s_B, il est déterminé à partir d'un facteur de qualité q_B, défini suivant l'heuristique suivante :

1. Si le bloc contient un pixel de teinte chair, $q_B = 80$
2. Sinon, si le bloc contient un pixel de la partie vitrée du véhicule, $q_B = 30$
3. Sinon, $q_B = 5$

De cette manière, si un pixel appartient à la fois à la classe des visages et de l'extérieur, on lui assigne le facteur de qualité des visages, qui est plus important que le facteur de qualité de l'extérieur. Ce cas intervient à chaque fois qu'un passager se trouve devant une vitre.

Finalement, s_B est calculé comme ci :

$$\begin{cases} \text{Si} q_B < 50 & s_B = 50/q_B \\ \text{Sinon} & s_B = 2 - q_B/50 \end{cases} \tag{7.7}$$

Les coefficients quantifiés sont alors utilisés pour reconstruire l'image dégradée, par transformée DCT inverse :

$$B_r(x,y) = \frac{1}{\sqrt{2N_B}} \sum_{i=0}^{N_B-1} \sum_{j=0}^{N_B-1} DCT_Q^B(i,j) \cos\left(\frac{(2x+1)i\pi}{2N_B}\right) \cos\left(\frac{(2y+1)j\pi}{2N_B}\right) \tag{7.8}$$

Un exemple de résultat du préfiltrage DCT est présenté sur la figure 7.3. Les facteurs de qualité q_B sont de 90 pour les pixels de teinte chair, 7 pour le paysage extérieur, et 4 pour l'intérieur du véhicule. Après compression JPEG, l'image a une taille de 21378 octets, contre 46383 octets pour la version non préfiltrée. Cela correspond à un gain de 54.9% sur la taille des données.

7.1.3.2 Le filtre gaussien

Le filtre gaussien effectue un lissage de l'image de façon très rapide. Il atténue les haute-fréquences de l'image, qui sont les fréquences dégradées en priorité lors de la quantification DCT de l'algorithme de compression JPEG, comme on peut le voir d'après les tables de quantification 7.5 et 7.6. Le paramètre qui varie en fonction des trois classes de pixels (visages, extérieur, intérieur) est simplement la variance de la gaussienne.



(a) (b)

FIG. 7.3 – Exemple de résultat du préfiltrage DCT. (a) :image originale, (b) : image préfiltrée

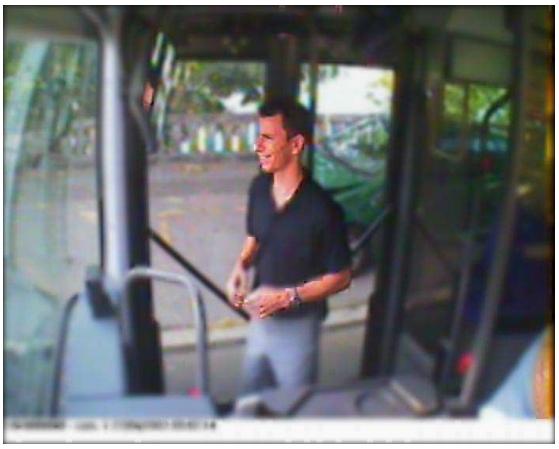


FIG. 7.4 – Exemple de résultat du préfiltrage gaussien

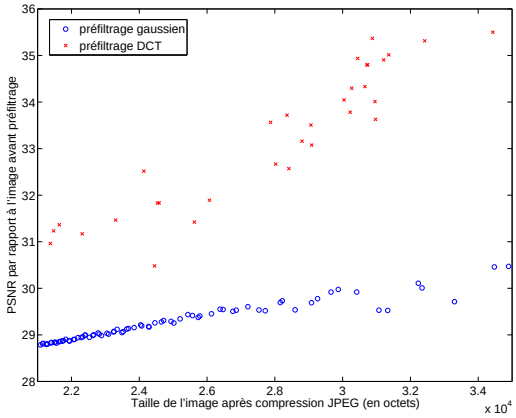


FIG. 7.5 – PSNR en fonction de la taille des données JPEG, pour les préfiltrages DCT et gaussien

La figure 7.4 illustre un exemple de résultat obtenu par filtrage gaussien. La variance de la gaussienne utilisée est de 15 pour le paysage extérieur, et 7 pour l'intérieur du véhicule. Les pixels du visages sont laissés inchangés. Les données JPEG obtenues après compression ont une taille de 32134 octets, ce qui représente un gain de 30.7%. L'image reste de qualité relativement correcte par rapport à l'original présenté figure 7.3(a), bien que les effets de bords entre les régions de classes différentes, lissées différemment, peuvent paraître visuellement gênants.

7.1.3.3 Comparaison entre les deux préfiltrages en compression d'image

Afin de pouvoir comparer les deux types de préfiltrage, nous étudions tout d'abord la qualité de l'image préfiltrée en fonction de taille du fichier résultant d'une compression JPEG. Un bon préfiltrage doit produire une image de bonne qualité, c'est à dire correctement exploitable pour une application de vidéosurveillance, et doit être compressée efficacement par un algorithme de type JPEG. Le résultat d'une compression JPEG est intéressant, bien qu'il ne s'agisse pas d'un algorithme de compression vidéo mais d'un algorithme de compression d'image. En effet, la plupart des algorithmes classiques de compression vidéo sont basés sur une quantification des coefficient de la DCT sur certaines images clés et d'un codage de la redondance temporelle entre images consécutives. L'étude de la taille des fichiers JPEG permet de quantifier les performances du préfiltrage indépendamment du codeur vidéo utilisé au final (MPEG1 2 ou 4) et de ses paramètres (espacement entre images *intra*).

La figure 7.5 représente l'évolution du PSNR entre l'image originale de la figure 7.2(a) et l'image préfiltrée en fonction de la taille du fichier JPEG. Les différents points de la figure sont obtenus pour des réglages différents des préfiltrages. Plus précisément, pour le préfiltrage DCT, nous faisons varier les facteurs de qualité des régions de l'intérieur et l'extérieur du véhicule, tout en maintenant un ratio constant entre ces deux facteurs



(a) (b)

FIG. 7.6 – Comparaison subjective des deux préfiltrages pour une même taille de fichier JPEG de 30ko. (a) : DCT. (b) : gaussien

de qualité. Pour le préfiltrage gaussien, c'est la variance des filtres pour les deux mêmes régions qui varie, toujours avec un ratio constant. Le PSNR n'est peut être pas la mesure la plus représentative de la qualité visuelle de l'image préfiltrée, mais les résultats obtenus nous confortent tout de même sur la supériorité du préfiltrage DCT par rapport au plus simple filtrage passe-bas gaussien.

Les images de la figure 7.6 sont les résultats des préfiltrages DCT et gaussien, produisant approximativement une même taille de JPEG de 30 kilo-octets. Bien que le filtrage gaussien produit une image tout à fait exploitable, la qualité visuelle est moins bonne que pour le préfiltrage DCT. Évidemment, il n'est pas très naturel d'utiliser un préfiltrage DCT pour notre application. En effet, d'après le schéma de la figure 7.1, l'image préfiltrée est fournie en entrée d'un encodeur vidéo matériel, que l'on choisit justement pour la rapidité de sa transformation en cosinus discrète. Les étapes de DCT et DCT inverses, réalisées de façon logicielle pendant le préfiltrage, sont donc redondantes. Néanmoins, les résultats obtenus pour ce préfiltrage DCT ne sont pas sans intérêt, car ils nous permettent d'envisager une autre approche pour la compression sélective, dans laquelle la quantification des coefficients de la DCT est contrôlée de manière adaptative directement à l'intérieur de l'encodeur vidéo. Ce regroupement des étapes de préfiltrage et de compression nécessite soit le remplacement de la compression matérielle par une compression logicielle, soit un compresseur matériel qui permet de contrôler la quantification de chaque bloc indépendamment.

7.1.3.4 Comparaison entre les deux préfiltrages en compression vidéo

Le gain en compression vidéo induit par le préfiltrage des images est aussi étudié. Il s'agit de comparer la taille des données vidéo compressées par un algorithme classique, ici h264 (norme aussi appelée MPEG-4 AVC), pour une séquence originale sans préfiltrage, et pour une séquence avec préfiltrage. De plus, les deux préfiltrages (DCT et gaussien)

Séquence	Taille (en octets)	Gain de taille
originale	766834	
DCT	955116	-24.55% (perte)
gaussien	228274	70.23%

Tab. 7.1 – Taille des données vidéo compressées par h264, avec et sans préfiltrage

sont considérés. Les paramètres des préfiltrages ont été choisis en fonction de la qualité visuelle résultante, de manière à s'approcher au plus des paramètres qui seraient utilisés en conditions réelles.

La figure 7.7 présente quelques images de la séquence vidéo de test utilisée pour la mesure des performances de la compression sélective. Cette séquence contient 50 images, et les carte de probabilité de teinte chair associées sont calculées. Après compression par l'algorithme h264 des trois séquences (originale, préfiltrée par DCT et préfiltrée par lissage gaussien), on obtient des données vidéos dont les tailles sont présentées dans le tableau 7.1. Bien entendu, les mêmes paramètres sont utilisés pour les trois séquences pour la compression h264. Ces résultats sont pour le moins surprenants car ils ne correspondent pas à nos attentes. Alors que le préfiltrage DCT était performant en compression d'image, il provoque ici une augmentation importante de la taille des données vidéo. On peut expliquer ce phénomène par une mauvaise estimation des vecteurs de mouvement lors de la compression h264, due à l'apparition d'effets de blocs importants à la suite du préfiltrage DCT. Par contre, le préfiltrage gaussien est très performant pour la compression sélective, avec un gain d'environ 70% sur la taille des données. Le lissage des images doit provoquer une meilleure redondance temporelle entre images consécutives, et permettre ainsi une bonne compression de l'information.

Il est à noter que les algorithmes de compression MPEG1 et MPEG2 ont aussi été testés, en remplacement de h264. Ils procurent des résultats similaires.

7.1.3.5 Influence de la variance du préfiltrage gaussien sur la taille de la vidéo compressée

Le préfiltrage gaussien est finalement plus intéressant que le préfiltrage par quantification DCT pour réduire la taille des données par une compression vidéo classique. Nous étudions l'influence du paramètre de variance du préfiltrage gaussien sur la taille de la vidéo compressée par l'algorithme h264.

On génère des séquences vidéo avec différentes variances pour le préfiltrage gaussien, controllées par un facteur de lissage f, prenant des valeurs entre 2 et 100 :

- La variance du lissage pour la région de l'exterieur du véhicule est $0.6f$.
- La variance du lissage pour la région de l'intérieur du véhicule est $1.8f$
- Les pixels de peau ne sont pas lissés

La figure 7.8 illustre le gain en taille que l'on peut obtenir en fonction du facteur de lissage.

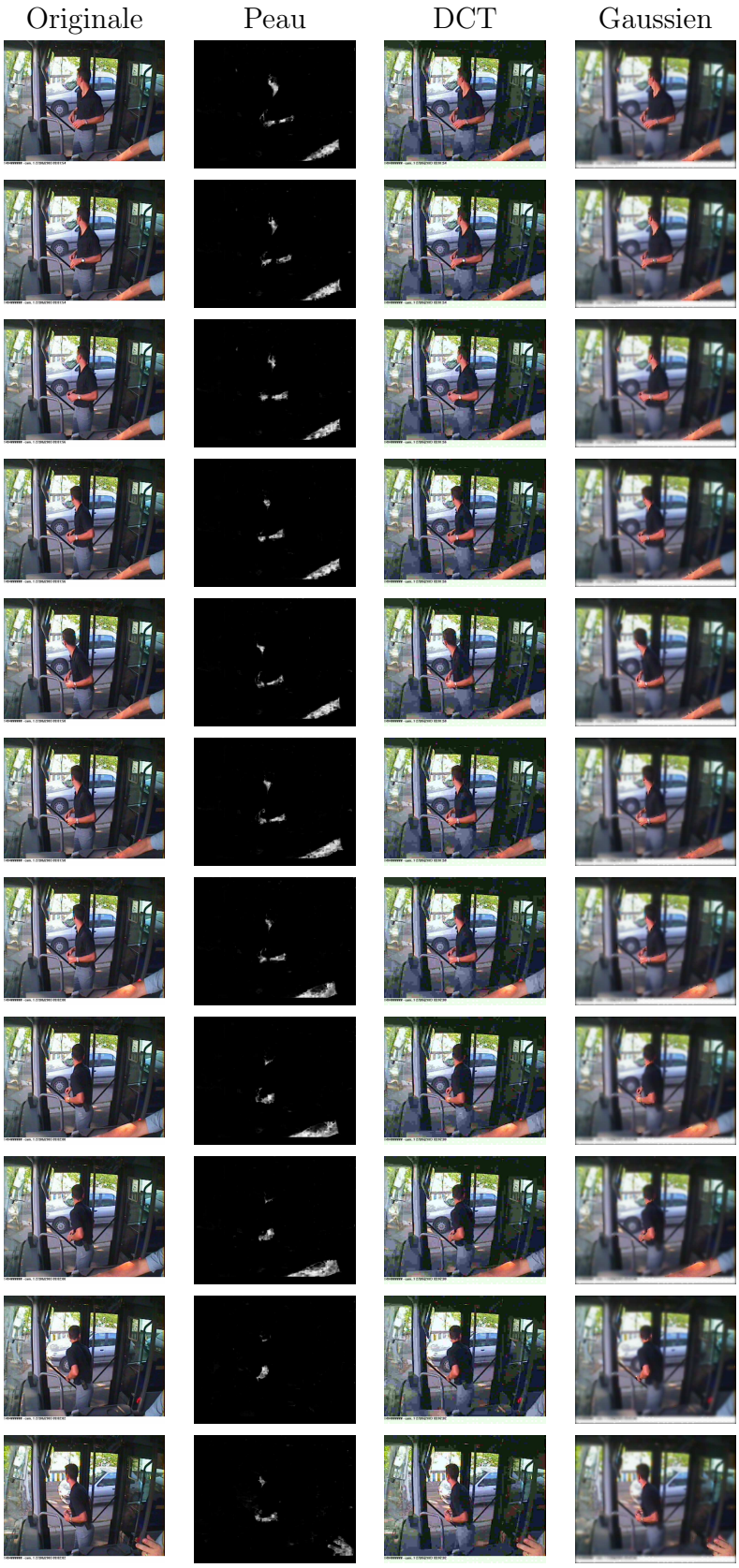


FIG. 7.7 – Séquence vidéo de test pour la compression sélective. Peau détectée et résultat des deux préfiltrages

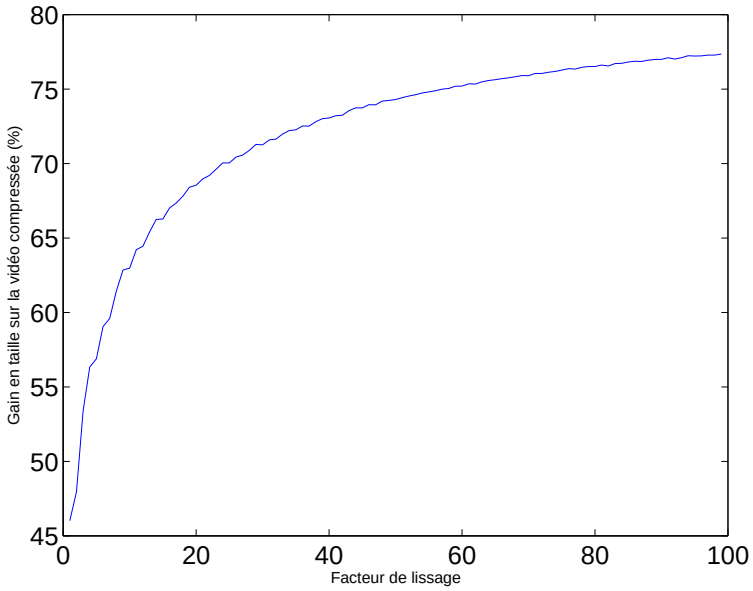


FIG. 7.8 – Gain en taille de la vidéo compressée par h264 par rapport à une compression sans préfiltrage, en fonction du facteur de lissage f

7.1.4 Conclusion sur la compression sélective

Les résultats obtenus en compression vidéo par le préfiltrage DCT montrent que cette solution n'est pas utilisable directement, sans modification du principe proposé initialement figure 7.1. Néanmoins, le gain de taille obtenu en compression JPEG nous conforte sur la possibilité d'obtenir de bons résultats en vidéo. De manière générale, un algorithme de compression vidéo génère un flot de données qui est constitué d'images clés compressées par un algorithme de type JPEG, de vecteurs de prédiction du mouvement inter-images, et d'un encodage de l'erreur entre les images réelles et les images prédites en mouvement. Après un préfiltrage DCT, la dégradation des images réduit la redondance temporelle dans la séquence, et l'erreur de prédiction prend alors une taille plus importante dans le flot de données. Cela explique les mauvaises performances du préfiltrage DCT en compression vidéo. Ce n'est pas le cas pour le préfiltrage gaussien, qui augmente au contraire la redondance temporelle dans la séquence. Le gain en compression est très bon si l'on utilise ce préfiltrage. Une modification que l'on peut envisager consisterait à réduire l'effet de blocs introduit par le préfiltrage DCT.

7.2 Comptage de personnes

L'application de comptage de personnes permet à l'exploitant d'un réseau de véhicules d'obtenir des statistiques sur le taux d'occupation d'une ligne en fonction du temps. Avec cette information, il lui est ensuite possible de réguler le nombre optimal de véhicules à

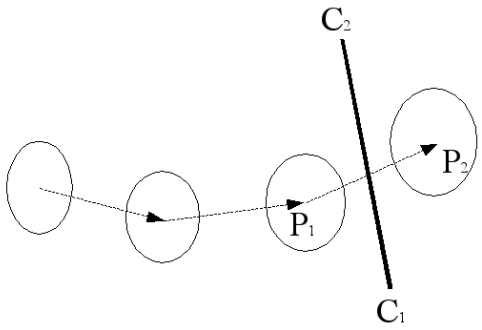


FIG. 7.9 – Comptage des visages passant un segment

déployer sur la ligne ainsi que les horaires de passage.

Le comptage est une application directe des méthodes développées pour la détection et le suivi des personnes au cours des séquences vidéo d'arrêts. L'analyse des trajectoires des passagers va permettre de détecter lorsqu'une personne entre ou sort du véhicule. Comme il a été montré précédemment que le suivi de personnes est sensible aux problèmes d'occultations, il est dangereux de se fier directement aux trajectoires résultantes pour réaliser le comptage. En effet, lorsqu'une personne est perdue par l'algorithme de suivi à cause d'une occultation qui n'a pas été gérée correctement, elle est généralement détectée de nouveau à l'image suivante, mais comme une personne différente. Il est important de ne pas compter deux fois la même personne dans ce cas.

7.2.1 Passage d'un segment

Une solution simple et efficace au problème des pertes de suivi de personnes consiste à compter le nombre de personnes passant par un endroit particulier dans l'image. Un segment imaginaire est défini dans l'image comme segment de comptage. Lorsqu'un visage traverse ce segment, on considère qu'une personne est montée ou descendue du véhicule, suivant le sens du passage. Avec les notations de la figure 7.9, on considère le passage d'un visage de la position P_1 à la position P_2, entre deux images consécutives. Ce visage traverse le segment défini par C_1 et C_2 si P_1 et P_2 sont tels que les inégalités suivantes sont vraies :

$$\begin{cases} \overrightarrow{C_1P_1}.\overrightarrow{C_1.C_2} > 0 \\ \overrightarrow{C_2P_1}.\overrightarrow{C_2C_1} > 0 \\ \overrightarrow{C_1P_2}.\overrightarrow{C_1.C_2} > 0 \\ \overrightarrow{C_2P_2}.\overrightarrow{C_2C_1} > 0 \\ \det(\overrightarrow{C_1P_1}, \overrightarrow{C_1C_2}) * \det(\overrightarrow{C_1P_2}, \overrightarrow{C_1C_2} < 0) \end{cases} \tag{7.9}$$

avec . l'opérateur de produit scalaire. D'autre part, le sens de traversée du segment est déterminé par le signe de $\det(\overrightarrow{C_1P_1}, \overrightarrow{C_1C_2})$.

L'avantage de cette méthode de comptage est qu'elle est simple à mettre en œuvre et très peu sensible aux pertes de suivi. En effet, il suffit que les visages soient suivis sur deux images consécutives, lors de la traversée du segment. Son principal inconvénient est qu'il est nécessaire de définir le segment de comptage manuellement, ce qui va à l'encontre de notre souhait d'avoir un système de surveillance dont l'installation est la plus simple et automatique possible.

7.2.2 Performances du comptage

Le comptage de personnes a été testé sur des séquences réelles de transport en commun, lors d'arrêts du véhicule. La figure 7.10 montre une séquence vidéo de résultats pour laquelle les images de chaque passage d'une personne à travers le segment de comptage ont été affichées. Ce segment est défini sur la droite de l'image, verticalement, juste à droite de la porte. Il est représenté par un segment tracé en blanc sur la figure. On observe que certaines personnes sont incorrectement détectées et suivies lors de leur passage à travers le segment. En particulier, les personnes des images 2, 5 et 8 n'ont pas pu être comptées.

Le taux moyen de comptage estimé est de 85% par rapport à la vérité terrain.

Ces performances sont assez faibles par rapport aux systèmes de comptage existants, utilisant d'autres moyens que l'analyse vidéo. Les résultats sont toutefois prometteurs, et démontrent la faisabilité d'un système de comptage de personnes se basant uniquement sur l'analyse des images acquises par le système de vidéosurveillance embarqué.

7.2.3 Conclusion sur l'application de comptage de personnes

L'application de comptage de personnes qui vient d'être présentée est un exemple simple qui met en évidence l'intérêt des méthodes d'analyse d'images qui ont été développées. La détection de personnes et le suivi inter-images lors d'un arrêt du véhicule sont effectivement des outils génériques dont le résultats doit être exploité par des applications telles que le comptage afin d'apporter des informations pertinentes à l'exploitant d'un réseau de véhicules. Pour l'instant, la cadence de traitement d'une séquence vidéo d'arrêt est encore trop faible, de lors d'une image toute les trois secondes, pour que le comptage soit réalisé en ligne. Les performances en comptage limitent aussi l'application à ne fournir qu'une approximation grossière du nombre réel de passagers.

7.3 Conclusions sur les applications développées

Deux applications relatives à l'analyse du contenu de la scène ont été présentées ici. Elles apportent des solutions, grâce aux outils d'analyse d'images introduits dans les chapitres 5 et 6, pour répondre en partie à deux besoins nouveaux d'un exploitant de réseau moderne de transport en commun.

Avec la compression sélective des séquences vidéo, il est possible de réduire de façon conséquente la taille des données vidéo tout en conservant une bonne qualité d'image



FIG. 7.10 – Comptage de personne sur une séquence d'arrêt

pour une exploitation correcte des images du point de vue de la vidéosurveillance. Le fort taux de compression obtenu permet d'envisager une transmission en direct de la vidéo par une liaison bas débit de type GSM ou GPRS, tout en conservant une cadence d'images suffisamment élevée. Nous avons envisagé deux types de préfiltrage pour la compression vidéo, à savoir un filtrage par quantification des coefficients DCT, de manière similaire à l'algorithme JPEG, et un filtrage gaussien classique. Les deux résultats sont intéressants car ils possèdent tous deux des qualités importantes. Le filtrage gaussien est réalisable en un temps très rapide, car le filtre est séparable, ce qui nous fait privilégier ce choix pour une implémentation de la compression sélective suivant le schéma défini figure 7.1. Le préfiltrage par DCT est quant à lui plus performant en terme de taux de compression, mais son implémentation serait plus pertinente si la quantification des coefficients de chaque bloc puisse être contrôlée directement au niveau du compresseur vidéo matériel.

Quant à l'application de comptage des personnes à un arrêt, elle répond à l'origine au besoin d'aide à l'exploitation d'un réseau de véhicules, en ouvrant la voie vers un système mesurant en temps réel le taux de remplissage des véhicules, à des fins d'optimisation du nombre de véhicules sur le réseau à chaque instant. Les performances obtenues sont pour l'instant trop faibles pour obtenir un comptage précis des passagers. Ils démontrent toutefois la possibilité de réaliser des applications d'analyse d'images d'assez haut niveau en environnement embarqué.

Conclusion et perspectives

Ce travail a été l'occasion d'explorer les diverses possibilités d'application de méthodes d'analyse d'images au contexte de vidéosurveillance embarquée dans les véhicules de transport en commun. Les spécificités des séquences vidéo traitées par rapport à celles issues d'un système de vidéosurveillance fixe classique ont nécessité le développement d'outils d'analyse particuliers. Un soin particulier a été donné aux aspects robustesse des méthodes ainsi qu'à leur implémentation efficace en terme de temps de calcul au sein d'un système embarqué.

Nous allons conclure ce manuscript en rappelant les principales contributions apportées au cours de ces travaux, soulignant l'originalité des méthodes proposées. Puis les perspectives de ce travail seront évoquées. Elles ne manquent pas, car le sujet est relativement vaste, et les améliorations à apporter sont donc nombreuses, ainsi que les applications envisageables.

Les apports

Les domaines de l'analyse d'images qui ont été abordés dans cette thèse sont nombreux. Il est donc naturel que les contributions qu'elle a apportée soient variées. Les résultats concernent deux catégories d'applications, qui sont d'une part l'analyse de l'état du système d'acquisition, et d'autre part l'analyse du contenu de la scène.

Parmi les travaux effectués sur l'analyse de l'état du système d'acquisition, nous retiendrons principalement deux points :

- Une caractérisation de la position du champ de vision d'une caméra a été définie à partir de l'extraction des contours stables. Cette information, rendant compte de la localisation des objets fixes de la scène, s'est révélée être suffisamment invariante aux diverses perturbations en illumination et mouvement, pour servir à des applications de détection de déplacement involontaire de la caméra, et d'aide à leur installation. L'originalité de ce travail se trouve dans la recherche d'une description invariante de la position de la caméra. La comparaison entre deux vues différentes se base ensuite sur cette description par les contours stables, et est effectuée par une mesure classique de distance globale à partir d'une transformée de distance.
- La définition d'une mesure de netteté originale a été présentée. Elle est basée sur l'estimation de la largeur des contours par l'adéquation d'un modèle de Gaussienne+bruit

au profil médian obtenu sur la norme du gradient en chaque point de contour. La mesure montre là aussi une bonne invariance au contenu de la scène et aux changements d'illumination. La seconde mesure de netteté proposée, présentée comme l'approximation de la première, considère le nombre de contours dans une image et le nombre de pixels appartenant aux contours, globalement sur l'image, afin d'en déduire une estimation de la largeur moyenne des contours. L'avantage de cette méthode est qu'elle peut se calculer très rapidement, sans dégrader la qualité de la mesure pour l'utilisation envisagée.

En ce qui concerne l'analyse du contenu de la scène, elle a principalement porté sur l'élaboration d'un algorithme complet de détection de personnes, depuis l'extraction des caractéristiques bas-niveau pertinentes jusqu'à la localisation des personnes dans l'image. Les principaux éléments développés dans ce domaine ont été les suivants :

- Une modélisation de l'arrière-plan de la scène lors d'un arrêt du véhicule a été proposée. Elle a été conçue de manière à prendre en compte les changements globaux d'illumination qui interviennent en particulier à cause des contrôles automatiques de balance des couleurs et de gain. Grâce à une représentation de l'image par la distribution des couleurs au voisinage de chaque pixel, la robustesse aux petits mouvements locaux est réalisée. L'algorithme de soustraction de fond associé, consistant en une analyse des résidus obtenus lors de l'estimation des paramètres de la transformation globale, donne de meilleurs résultats qu'une soustraction de fond classique par mélanges de Gaussiennes.
- Un modèle de personne adapté à notre problématique, combinant les informations bas-niveau de teinte chair et d'avant-plan, a été décrit et évalué sur des séquences vidéo en conditions réelles. La combinaison des informations bas-niveau est réalisée directement au sein de la définition du modèle de personne, qui spécifie les relations spatiales entre les deux sources. Le modèle est assez général pour permettre une détection robuste aux différentes poses que peuvent prendre les personnes dans notre contexte. Il est en même temps assez précis dans sa représentation d'une personne pour que le taux de fausses détections soit très faible. L'estimation des paramètres du modèle par un échantillonnage de Monte-Carlo suivi d'une sélection des échantillons les plus probables donne de très bons résultats sur les séquences considérées, avec un taux de bonnes détections de 97% pour 4% de fausses alarmes. L'algorithme de suivi par similarité en apparence, qui se base sur cette détection de personnes et d'une description concise des couleurs de chaque personne, permet d'envisager une application de comptage de personnes.
- Grâce à l'extraction des informations bas-niveau de teinte chair et d'appartenance à une vitre, une application de compression vidéo sélective a pu être réalisée. La segmentation de l'image en trois classes, correspondant aux zones de peau, à l'intérieur et à l'extérieur du bus, contrôle les paramètres d'un préfiltrage de l'image. Les deux types de préfiltrage considérés ont été une quantification des coefficients DCT, de manière équivalente à JPEG, et un préfiltrage Gaussien. Ce préfiltrage Gaussien est

choisi dans l'implémentation de la compression vidéo sélective, pour ses performances supérieures en terme de débit-distortion, qui permettent d'envisager sont utilisation pour la transmission des données vidéo par une liaison à très bas débit.

Les perspectives

Les perspectives de cette thèse sont nombreuses. Elles portent soit sur l'amélioration des algorithmes proposés en terme de robustesse et de vitesse de calcul, soit sur le développement d'autres applications relatives aux transports en commun. L'industrialisation des méthodes développées est aussi une étape importante du projet qui doit être finalisée.

Concernant les améliorations possibles, l'algorithme de modélisation de l'arrière-plan et d'extraction des pixels d'avant-plan est probablement celui qui mérite le plus d'attention. La transformation temporelle des vecteurs d'arrière-plan a été supposée globale et linéaire, ce qui empêche la prise en compte de certains changements de la scène comme l'apparition de reflets sur les parties vitrées du véhicule. Il serait intéressant de faire évoluer l'algorithme proposé vers une prise en compte de changements d'illumination plus complexes. De plus, les vecteurs de caractéristiques en chaque pixel contiennent des statistiques qui supposent une distribution monomodale des couleurs dans chaque fenêtre d'analyse. Cette hypothèse n'est pas toujours vérifiée, notamment aux alentours des points de contours, où la présence de deux couleurs implique une distribution bimodale des couleurs. Par conséquent, le modèle utilisé pour l'arrière-plan est mal adapté aux zones de l'image contenant de forts contours. L'intégration d'un modèle plus complexe pour la distribution des pixels, comme par exemple un mélange de Gaussiennes, constituerait une perspective intéressante.

Les recherches futures doivent aussi porter sur des améliorations algorithmiques permettant un calcul plus rapide pour la modélisation de l'arrière-plan, ainsi que pour la détection de personnes. Cette dernière pourrait bénéficier d'une méthode d'estimation plus rapide des paramètres. L'estimation par échantillonnage de Monte-Carlo utilisée actuellement pourrait en effet être remplacée par un échantillonnage itératif de type filtrage particulaire, de façon à ce que le nombre d'échantillons nécessaire soit moins important. On bénéficierait dans ce cas des avantages d'une optimisation locale des paramètres, permettant de localiser plus précisément les personnes.

En terme d'applications, des méthodes d'indexation automatique d'évènements sont envisagées pour la relecture des enregistrements vidéo à partir des méthodes de détection et suivi de personnnes développées au cours de cette thèse. Il est intéressant de pouvoir parcourir les enregistrements vidéos avec l'option de se déplacer directement sur certains évènements comme l'ouverture de la porte, ou la montée d'un passager. Les outils d'analyse d'images développés au cours de cette thèse peuvent amener au développement d'une telle application. Le contexte des véhicules de transport en commun permet d'envisager à partir

de ces travaux d'autres types d'applications comme la surveillance d'une zone particulière du véhicule ou la détection d'objets abandonnés.

En ce qui concerne l'industrialisation des méthodes développées, l'évolution constante des algorithmes et les difficultés auxquelles nous avons été confrontés pour obtenir des retours sur les systèmes installés chez les clients ont rendu délicates les phases de test et de validation. Il est notamment indispensable de valider nos algorithmes dans des conditions réelles, avec un jeu de données plus conséquent et dans un environnement embarqué contraignant. Une analyse plus poussée du comportement des différents algorithmes sur une base de test de taille réelle pourrait aussi permettre l'apprentissage et la justification de quelques paramètres qui ont été fixés de manière empirique. Les expériences réalisées en laboratoire sur nos séquences vidéo en conditions réelles et leurs résultats nous confortent tout de même sur les bonnes performances que l'on peut envisager suite à une véritable industrialisation de ce travail.

Liste des publications

S. Harasse, L. Bonnaud et M. Desvignes. - Background estimation for dynamic video scenes. In : *Proceedings of the 32nd Annual Conference of the IEEE Industrial Electronics Society, IECON*, Paris, France, november 2006.

S. Harasse, L. Bonnaud et M. Desvignes. - A human model for detecting people in video from low level features. In : *Proceedings of the IEEE International Conference on Image Processing, ICIP*, Atlanta, october 2006.

S. Harasse, L. Bonnaud et M. Desvignes. - Human model for people detection in dynamic scenes. In : *Proceedings of the International Conference of Pattern Recognition, ICPR*, pp. 335-354. - Hong Kong, august 2006.

S. Harasse, L. Bonnaud et M. Desvignes. - Mutliple faces tracking using local statistics. In : *Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis, ISPA*, pp. 18-23. - Zagreb, Croatia, september 2005.

S. Harasse, L. Bonnaud et M. Desvignes. - Finding people in video streams by statistical modeling. In : *Proceedings of the 3rd ICAPR International Conference on Advances in Pattern Recognition*, pp. 608-617. - Bath, UK, august 2005.

S. Harasse, L. Bonnaud et M. Desvignes. - People counting in transport vehicles. In : *Proceedings of the International Conference on Pattern Recognition and Computer Vision*, pp. 221-224. - Istanbul, Turkey, february 2005.

S. Harasse, M. Desvignes, L. Bonnaud et A. Caplier. - Detection d'anomalies pour des cameras mobiles. In : *CORESA*, pp. 77-80. - Lille, France, may 2004.

S. Harasse, L. Bonnaud, A. Caplier et M. Desvignes. - Automated camera dysfunctions detection. In : *Proceedings of the IEEE Southwest Symposium on Image Analysis and Interpretation*, pp. 36-40. - Lake Tahoe, Nevada, USA, march 2004.

Bibliographie

[AK03] Haikel Salem Alhichri et Mohamed Kamel. Virtual circles : a new set of features for fast image registration. *Pattern Recogn. Lett.*, 24(9-10) :1181–1190, 2003.

[Anu70] P.E. Anuta. Spatial registration of multispectral and multitemporal digital imagery using fast fourier transform techniques. *IEEE Trans. on Geoscience Electronics*, 8 :353–368, octobre 1970.

[AT06] Ankur Agarwal et Bill Triggs. Recovering 3d human pose from monocular images. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 28(1), janvier 2006.

[BAHH92] James R. Bergen, P. Anandan, Keith J. Hanna et Rajesh Hingorani. Hierarchical model-based motion estimation. In *ECCV '92 : Proceedings of the Second European Conference on Computer Vision*, pages 237–252, London, UK, 1992. Springer-Verlag.

[BDH+76] JF Brenner, BS Dew, JB Horton, T King, PW Neurath et WD Selles. An automated microscope for cytologic research a preliminary evaluation. *Journal of Histochemistry and Cytochemistry*, 24(1) :100–111, 1976.

[Ber98] Rikard Berthilsson. Affine correlation. *Proceedings of the International Conference on Pattern Recognition*, 02 :1458, 1998.

[BH72] E. I. Barnea et H.F.Silverman. A class of algorithms for fast digital image registration. *IEEE Transactions on Computers*, 21 :179–186, 1972.

[BK89] R. Bajcsy et S. Kovacic. Multiresolution elastic matching. *Computer Vision, Graphics and Image Processing*, 46 :1–21, 1989.

[Bor84] G. Borgefors. Distance transformation in arbitrary dimensions. *Computer Vision, Graphics, and Image Processing*, 27 :321–345, 1984.

[Bor88] Gunilla Borgefors. Hierarchical chamfer matching : A parametric edge matching algorithm. *IEEE Trans. Pattern Anal. Mach. Intell.*, 10(6) :849–865, 1988.

[BS97] Debashis Bhattacharya et Satyabroto Sinha. Invariance of stereo images via the theory of complex moments. *Pattern Recognition*, 30(9) :1373–1386, 1997.

[BTBW77] H. G. Barrow, J. M. Tenenbaum, R. C. Bolles et H. C. Wolf. Parametric correspondence and chamfer matching : Two new techniques for image matching. In *Proc. of the 5th IJCAI*, pages 659–663, Cambridge, MA, 1977.

[Cam80] N. A. Campbell. Robust procedures in multivariate analysis. i : Robust variance estimation. *Applied Statistics*, 29(3) :231–237, 1980.

[Can86] J Canny. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8(6) :679–698, 1986.

[CBBS94] James L. Crowley, Jean Marc Bedrune, Morten Bekker et Michael Schneider. Integration and control of reactive visual processes. In *ECCV '94 : Proceedings of the third European conference on Computer Vision (Vol. II)*, pages 47–58, Secaucus, NJ, USA, 1994. Springer-Verlag New York, Inc.

[CN98] D. Chai et K. N. Ngan. Locating facial region of a head-and-shoulders color image. In *FG '98 : Proceedings of the 3rd. International Conference on Face & Gesture Recognition*, page 124, Washington, DC, USA, 1998. IEEE Computer Society.

[Cot05] Jean-Mattieu Cotton. Compression sélective pour le stockage vidéo. Technical report, ENSIMAG-TELECOM, 2005.

[CZ03] D. Capel et A. Zisserman. Computer vision applied to super resolution. *IEEE Signal Processing Magazine*, 20(3) :75–86, mai 2003.

[DCWS03] Gianfranco Doretto, Alessandro Chiuso, Ying Nian Wu et Stefano Soatto. Dynamic textures. *Int. J. Comput. Vision*, 51(2) :91–109, 2003.

[DF01] Frédéric Devernay et Olivier Faugeras. Straight lines have to be straight : automatic calibration and removal of distortion from scenes of structured enviroments. *Mach. Vision Appl.*, 13(1) :14–24, 2001.

[DK97] X. Dai et S. Khorram. Development of a feature-based approach to automated image registration for multitemporal and multisensor remotely sensed imagery. In *IEEE Internat. Geoscience and Remote Sensing Symposium Proc. Remote Sensing*, volume 1, pages 243–245, 1997.

[Don82] D. L. Donoho. *Breakdown properties of multivariate location estimators*. PhD thesis, Harvard University, 1982.

[EHD00] A. Elgammal, D. Harwood et L. S. Davis. Non-parametric model for background subtraction. In *6th European Conference on Computer Vision*, 2000.

[FC97] L. Fonseca et M. Costa. Automatic registration of satellite images. In *Brazilian Symposium on Graphic Computation and Image Processing*, pages 219–226, 1997.

[Fel01] P. Felzenszwalb. Learning models for object recognition. In *Computer Vision and Pattern Recognition*, volume 1, pages 56–62, 2001.

[FGH01] S. Fortune, P. Gough et M. Hayes. Statistical autofocus of synthetic aperture sonar images using image contrast optimisation. In *IGARSS*, 2001.

[Fit01] A. W. Fitzgibbon. Simultaneous linear estimation of multiple view geometry and lens distortion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2001.

[FPZ03] R. Fergus, P. Perona et A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 264–271, Madison, Wisconsin, juin 2003.

[FR97] Nir Friedman et Stuart Russell. Image segmentation in video sequences : A probabilistic approach. In *Annual Conference on Uncertainty in Artificial Intelligence*, pages 175–181, 1997.

[FS95] Yoav Freund et Robert E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. In *European Conference on Computational Learning Theory*, pages 23–37, 1995.

[FWKP05] Charles Florin, James Williams, Ali Khamene et Nikos Paragios. Registration of 3d angiographic and x-ray images using sequential monte carlo sampling. In *CVBIA*, pages 427–436, 2005.

[Gav00] D. M. Gavrila. Pedestrian detection from a moving vehicle. In *ECCV*, pages 37–49, 2000.

[GCG+96] Hans Peter Graf, Eric Cosatto, Dave Gibbon, Michael Kocheisen et Eric Petaja. Multi-modal system for locating heads and faces. In *Proceedings of the 2nd International Conference on Automatic Face and Gesture Recognition*, page 88, Washington, DC, USA, 1996. IEEE Computer Society.

[GCPC95] H.P. Graf, T. Chen, E. Petajan et E. Cosatto. Locating faces and facial parts. In *First International Workshop Automatic Face and Gesture Recognition*, pages 41–46, 1995.

[Gro87] P. Grossman. Depth from focus. *Pattern Recognition Letters*, 5 :63–69, 1987.

[GS99] W. E. L. Grimson et C. Stauffer. Adaptive background mixture models for real time tracking. In *CVPR*, 1999.

[GSC98] Venu Govindu, Chandra Shekhar et Rama Chellappa. Using geometric properties for correspondence-less image alignment. In *Proceedings of the International Conference on Pattern Recognition*, volume 1, pages 37–41, Washington, DC, USA, 1998. IEEE Computer Society.

[GSP86] A. Goshtasby, G. C. Stockman et C. V. Page. A region-based approach to digital image registration with subpixel accuracy. *IEEE Transactions on Geoscience and Remote Sensing*, 24 :390–399, mai 1986.

[GSRL98] W. E. L. Grimson, C. Stauffer, R. Romano et L. Lee. Using adaptive tracking to classify and monitor activities in a site. In *CVPR '98 : Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, page 22, Washington, DC, USA, 1998. IEEE Computer Society.

[Har02] Michael Harville. A framework for high-level feedback to adaptive, per-pixel, mixture-of-gaussian background models. In *ECCV '02 : Proceedings of the 7th European Conference on Computer Vision-Part III*, pages 543–560, London, UK, 2002. Springer-Verlag.

[HF93] N. Hanaizumi et S. Fujimur. An automated method for registration of satellite remote sensing images. In *International Geoscience and Remote Sensing Symposium*, volume 3, pages 1348–1350, Tokyo, Japan, août 1993.

[HHD00] I. Haritaoglu, D. Harwood et L.S. Davis. W4 : Real-time surveillance of people and their activities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8), août 2000.

[HJP92] Y.C. Hsieh, D.M. McKeown Jr. et F.P. Perlant. Performance evaluation of scene registration and stereo matching for cartographic feature extraction. *IEEE Transactions PAMI*, 14 :214–238, février 1992.

[HKR93] D. Huttenlocher, D. Klanderman et A. Rucklige. Comparing images using the Hausdorff distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(9) :850–863, septembre 1993.

[HLFK96] J-W. Hsieh, H-Y.M. Liao, K-C. Fan et M-T. Ko. A fast algorithm for image registration without predetermining correspondences. *Proceedings of the International Conference on Pattern Recognition*, 01 :765, 1996.

[HW77] P. W. Holland et R. E. Welsch. Robust regression using iteratively reweighted least squares. *Commun. Stat*, A6 :813–828, 1977.

[IB95] S. S. Intille et A. F. Bobick. Closed-world tracking. In *ICCV '95 : Proceedings of the Fifth International Conference on Computer Vision*, page 672, Washington, DC, USA, 1995. IEEE Computer Society.

[IB98] Michael Isard et Andrew Blake. Condensation – conditional density propagation for visual tracking. *International Journal of Computer Vision*, 29(1) :5–28, 1998.

[IDB97] S. S. Intille, J. W. Davis et A. F. Bobick. Real-time closed-world tracking. In *CVPR '97 : Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)*, page 697, Washington, DC, USA, 1997. IEEE Computer Society.

[IP91] Michal Irani et Shmuel Peleg. Improving resolution by image registration. *CVGIP : Graph. Models Image Process.*, 53(3) :231–239, 1991.

[KHM+00] John Krumm, Steve Harris, Brian Meyers, Barry Brumitt, Michael Hale et Steve Shafer. Multi-camera multi-person tracking for easyliving. In *VS '00 : Proceedings of the Third IEEE International Workshop on Visual Surveillance (VS'2000)*, page 3, Washington, DC, USA, 2000. IEEE Computer Society.

[KI01] R. Khalaf et S.S. Intille. Improving multiple people tracking using temporal consistency. Technical report, MIT Dept. of Architecture House_n Project, 2001.

[KL94] Y. Kwon et N. Da Vitoria Lobo. Face detection using templates. In *Proceedings of the International Conference on Pattern Recognition*, Israel, octobre 1994.

[Kro87] Eric Krotkov. Focusing. *Intl. Journal of Computer Vision*, 1(3) :223–237, octobre 1987. Reprinted in Physics-Based Vision, Jones and Bartlett, 1992.

[KS90] M. Kirby et L. Sirovich. Application of the karhunen-loeve procedure for the characterization of human faces. *IEEE Trans. Pattern Anal. Mach. Intell.*, 12(1) :103–108, 1990.

[KWM94] Dieter Koller, Joseph Weber et Jitendra Malik. Robust multiple car tracking with occlusion reasoning. In *ECCV (1)*, pages 189–196, 1994.

[LBP95] T.K. Leung, M. C. Burl et P. Perona. Finding faces in cluttered scenes using random labeled graph matching. In *Fifth Intl. Conf. on Comp. Vision*, juin 1995.

[Leh98] Thomas M. Lehmann. A two-stage algorithm for model-based registration of medical images. In *Proceedings of the International Conference on Pattern Recognition*, volume 1, page 344, Washington, DC, USA, 1998. IEEE Computer Society.

[Lev44] K. Levenberg. A method for the solution of certain problems in least squares. *Quart. Appl. Math.*, 2 :164–168, 1944.

[LG02] L. Lee et W. Grimson. Gait analysis for recognition and classification. In *the IEEE Conference on Face and Gesture Recognition*, pages 155–161, 2002.

[LKP92] Stan Z. Li, Josef Kittler et Maria Petrou. Matching and recognition of road networks from aerial images. In *ECCV*, pages 857–861, 1992.

[LMM95] H. Li, B.S. Manjunath et S.K. Mitra. A contour-based approach to multisensor image registration. *IEEE Trans on Image Processing*, 4(3) :320–334, mars 1995.

[Low99] David G. Lowe. Object recognition from local scale-invariant features. In *Proc. of the International Conference on Computer Vision ICCV, Corfu*, pages 1150–1157, 1999.

[LSS05] Bastian Leibe, Edgar Seemann et Bernt Schiele. Pedestrian detection in crowded scenes. In *CVPR '05 : Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1*, pages 878–885, Washington, DC, USA, 2005. IEEE Computer Society.

[LY87] Maylor K. Leung et Yee-Hong Yang. Human body motion segmentation in a complex scene. *Pattern Recogn.*, 20(1) :55–64, 1987.

[Mar76] R. Maronna. Robust m-estimators of multivariate location and scatter. *The Annals of Statistics*, 4 :51–67, 1976.

[MB00] John MacCormick et Andrew Blake. A probabilistic exclusion principle for tracking multiple objects. *Int. J. Comput. Vision*, 39(1) :57–71, 2000.

[MBM97] B. Marcel, M. Briot et R. Murrieta. Calcul de translation et rotation par la transformation de fourier. *Traitement du Signal*, 14(2) :135–149, 1997.

[MDWE02] P. Marziliano, Frederic Dufaux, Stefan Winkler et Touradj Ebrahimi. A no-reference perceptual blur metric. In *the International Conference on Image Processing*, volume 3, pages 57–60, 2002.

[MMPR03] A. Monnet, A. Mittal, N. Paragios et V. Ramesh. Background modeling and subtraction of dynamic scenes. In *International Conference on Computer Vision*, pages 1305–1312, 2003.

[MP04] A. Mittal et N. Paragios. Motion-based background subtraction using adaptive kernel density estimation. In *Computer Vision and Pattern Recognition*, juin 2004.

[MPP01] Anuj Mohan, Constantine Papageorgiou et Tomaso Poggio. Example-based object detection in images by components. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(4) :349–361, 2001.

[MRG98] Stephen J. McKenna, Yogesh Raja et Shaogang Gong. Object tracking using adaptive color mixture models. In *ACCV (1)*, pages 615–622, 1998.

[MSC96] B. Manjunath, C. Shekhar et R. Chellappa. A new approach to image feature detection with applications. *Pattern Recognition*, 31 :627–640, 1996.

[MSZ04] Krystian Mikolajczyk, Cordelia Schmid et Andrew Zisserman. Human detection based on a probabilistic assembly of robust part detectors. In *European Conference on Computer Vision*, volume I, pages 69–81, 2004.

[Mur92] F. Murtagh. A feature-based o(n^2) approach to point pattern matching. In *International Conference on Pattern Recognition*, volume 2, pages 174–177, 1992.

[MvdEV96a] J. B. Antoine Maintz, Petra A. van den Elsen et Max A. Viergever. Evaluation of ridge seeking operators for multimodality medical image matching. *IEEE Trans. Pattern Anal. Mach. Intell.*, 18(4) :353–365, 1996.

[MvdEV96b] J.B.A. Maintz, P.A. van den Elsen et M.A. Viergever. Comparison of edge-based and ridge-based registration of CT and MR brain images. *Medical Image Analysis*, 1(2) :151–161, 1996.

[MW87] Henri Maître et Yifeng Wu. Improving dynamic programming to solve image registration. *Pattern Recogn.*, 20(4) :443–461, 1987.

[NA74] K. R. Rao N. Ahmed, T. Natarajan. Discrete cosine transform. *IEEE Trans. Computer*, 23(1) :90–93, janvier 1974.

[Nob88] J. Alison Noble. Finding corners. *Image Vision Comput.*, 6(2) :121–128, 1988.

[NPA01] Kuang-Chern Ng, Aun Neow Poo et Marcelo H. Ang. Practical issues in pixel-based autofocusing for machine vision. In *ICRA*, pages 2791–2796, 2001.

[OPS+97] M. Oren, C. Papageorgiou, P. Shinha, E. Osuna et T. Poggio. A trainable system for people detection. In *Image Understanding Workshop*, pages 207–214, 1997.

[Pen87] A. P. Pentland. A new sense for depth of field. *IEEE Trans. Pattern Anal. Mach. Intell.*, 9(4) :523–531, 1987.

[PK02] Janez Pers et Stanislav Kovacic. Nonparametric, model-based radial lens distortion correction using tilted camera assumption. In *Proceedings of the Computer Vision Winter Workshop 2002*, pages 286–295, Bad Aussee, Austria, février 2002.

[PMS94] A. Pentland, B. Moghaddam et T. Starner. View-based and modular eigenspaces for face recognition. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR'94)*, Seattle, WA, juin 1994.

[Pra74] W. Pratt. Correlation techniques of image registration. *IEEE Transactions on Aerospace and Electronic Systems*, 10 :353–358, 1974.

[PWHR98] P. Jonathon Phillips, Harry Wechsler, Jeffrey Huang et Patrick J. Rauss. The feret database and evaluation procedure for face-recognition algorithms. *Image Vision Comput.*, 16(5) :295–306, 1998.

[QSM98] Richard J. Qian, M. Ibrahim Sezan et Kristine E. Matthews. A robust real-time face tracking algorithm. In *Proceedings of the International Conference on Image Processing*, volume 1, pages 131–135, 1998.

[RBK98] Henry Rowley, Shumeet Baluja et Takeo Kanade. Neural network-based face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(1) :23–38, janvier 1998.

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

[RK76] A. Rosenfeld et A. Kak. *Digital picture processing.* Academic Press, New York, 1976.

[RLM05] Belaroussi R., Prevost L. et Milgram. Combining model-based classifiers for face localization. In *IAPR Conference on Machine Vision Applications, MVA2005*, 2005.

[Rou85] P. Rousseeuw. Multivariate estimation with high breakdown point. *Mathematical Statistics and Applications*, B :283–297, 1985.

[RRPP02] F. Rooms, M. Ronsse, A. Pizurica et W. Philips. Psf estimation with applications in autofocus and image restoration. In *Proceedings of the 3rd IEEE Benelux Signal Processing Symposium (SPS)*, pages 13–16, 2002.

[SC00] K. Schwerdt et J. Crowley. Robust face tracking using color. In *the 4th International Conference on Automatic Face and Gesture Recognition*, pages 90–95, 2000.

[SF96] David Saxe et Richard Foulds. Toward robust skin identification in video images. In *FG '96 : Proceedings of the 2nd International Conference on Automatic Face and Gesture Recognition (FG '96)*, page 379, Washington, DC, USA, 1996. IEEE Computer Society.

[SHW98] Q. B. Sun, W. M. Huang et J. K. Wu. Face detection based on color and local symmetry information. In *FG '98 : Proceedings of the 3rd. International Conference on Face & Gesture Recognition*, page 130, Washington, DC, USA, 1998. IEEE Computer Society.

[SI95] Ashok Samal et Prasana A. Iyengar. Human face detection using silhouettes. *IJPRAI*, 9(6) :845–867, 1995.

[Sim96] A. Simper. Correcting general band-to-band misregistrations. *Proceedings of the International Conference on Image Processing*, B :597–600, 1996.

[SKB82] George C. Stockman, S. Kopstein et S. Benett. Matching Images to Models for Registration and Object Detection via Clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-4(3) :229–241, 1982.

[SM04] Gregory Shakhnarovich et Baback Moghaddam. Face recognition in subspaces, 2004.

[SN00] R. Swaminathan et S.K. Nayar. Non-Metric Calibration of Wide-Angle Lenses and Polycameras. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(10) :1172 – 1178, octobre 2000.

[SO95] Y. Sumi et Y. Ohta. Detection of face orientation and facial components using distributed appearance modeling. In *International Workshop on Automatic Face and Gesture Recognition*, pages 245–249, 1995.

[SOCM01] H. S. Stone, M. T. Orchard, E. C. Chang et S. A. Martucci. A fast direct fourier-based algorithmfor subpixel registration of images. *IEEE Transactions on Geoscience and Remote Sensing*, 39(10) :2235–2243, octobre 2001.

[SPM97] Dongseok Shin, J.K. Pollard et J.-P. Muller. Accurate geometric correction of atsr images. *IEEE Transactions on Geoscience and Remote Sensing*, 35(4) :997–1006, 1997.

[SSS+96] S. Satoh, Toshio Sato, Michael Smith, Y. Nakamura et Takeo Kanade. Name-it : Naming and detecting faces in news video. *Network-Centric Computing (NCC) Special Issue*, 1996.

[SSV+97] A. Santos, C. Ortiz De Solorzano, J.J. Vaqueros, J.M. Peena, N. Malpica et F. Del Pozo. Evaluation of autofocus functions in molecular cytogenetic analysis. *Journal of Microscopy*, 188(3) :264–272, décembre 1997.

[Sta81] W.A. Stahel. Breakdown of covariance estimators. Research Report, 31, Fachgruppe fur Statistik, ETH, Zurich, 1981.

[Ség04] Renaud Séguier. A very fast adaptive face detection system. In *International Conference on Visualization, Imaging, and Image Processing (VIIP 2004)*, 2004.

[TG04] T. Tuytelaars et L. Van Gool. Matching widely separated views based on affinely invariant neighbourhoods. *IJCV*, 59(1) :51–85, 2004.

[THC03] D. Tsishkou, M. Hammami et Liming Chen. Face detection in video using combined data-mining and histogram based skin-color model. In *Proceedings of the 3rd International Symposium on Image and Signal Processing and Analysis, ISPA*, volume 1, pages 500–503, 2003.

[TJ89] J. Ton et A. K. Jain. Registering landsat images by point matching. *IEEE Trans. Geosci. Remote Sensing*, 27(5) :642–651, 1989.

[TKBM99] Kentaro Toyama, John Krumm, Barry Brumitt et Brian Meyers. Wallflower : Principles and practice of background maintenance. In *ICCV (1)*, pages 255–261, 1999.

[TLX00] X. Tang, P. L'Hostis et Y. Xiao. An auto-focusing method in microscopic testbed for optical discs. *Journal of Research of the National Institute of Standards and Technology*, 105(4), juillet 2000.

[TU96] P. Thevenaz et M. Unser. A pyramid approach to sub-pixel image fusion based on mutual information. In *Proceedings of the International Conference on Image Processing*, pages 265–268, 1996.

[TU98] P. Thévenaz et M. Unser. An efficient mutual information optimizer for multi-resolution image registration. In *Proceedings of the International Conference on Image Processing*, volume 1, pages 833–837, Chicago IL, USA, octobre 1998.

[VB97] N. Vujovic et D. Brzakovic. Establishing the correspondence between control points in pairs of mammographic images. *IEEE Trans. Image Processing*, 6(10) :1388–1399, 1997.

[VJ04] Paul Viola et Michael J. Jones. Robust real-time face detection. *Int. J. Comput. Vision*, 57(2) :137–154, 2004.

[VJS03] P. Viola, M. Jones et D. Snow. Detecting pedestrians using patterns of motion and appearance. Technical report, Mitsubishi Electric Research Lab, 2003.

[Vol88] D. Vollath. The influence of the scene parameters and of noise on the behaviour of automatic focusing algorithms. *Journal of Microscopy*, 151 :133–146, 1988.

[VSV03] P. Vandewalle, S. Susstrunk et M. Vetterli. Superresolution images reconstructed from aliased images. In *SPIE/IS&T Visual Communication and Image Processing Conference*, pages 1398–1405, juillet 2003.

[VSV05] Patrick Vandewalle, Sabine Süsstrunk et Martin Vetterli. A Frequency Domain Approach to Registration of Aliased Images with Application to Super-Resolution. *EURASIP Journal on Applied Signal Processing (special issue on Super-resolution)*, 2005.

[VWI95] P. Viola et W. Wells III. Alignment by maximization of mutual information. In *ICCV 95*, pages 16–23, 1995.

[VZB98] A. S. Vasileisky, B. Zhukov et M. Berger. Automated image co-registration based on linear feature recognition. In *Proc. 2nd Conf. Fusion of Earth Data*, pages 59–66, Sophia Antipolis, France, 1998.

[WADP97] Christopher Richard Wren, Ali Azarbayejani, Trevor Darrell et Alex Pentland. Pfinder : Real-time tracking of the human body. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7) :780–785, 1997.

[Wei94] Tse-Chung Wei. *Three dimensional machine vision using image defocus*. PhD thesis, Dept. of Electrical Engineering, SUNY at Stony Brook, 1994. Adviser-Murali Subbarao.

[WHKG90] J. Wiklund, L. Haglund, H. Knutsson et G. H. Granlund. Time Sequence Analysis Using Multi-Resolution Spatio-Temporal Filters. In V. Cappellini, editor, *Time-Varying Image Processing and Moving Object Recognition*, pages 258–265. Elsevier Science Publishers, 1990.

[WS77] P. Van Wie et M. Stein. A landsat digital image rectification system. *IEEE Trans. on Geoscience Electronics*, 15(3) :130–137, juillet 1977.

[WSYR83] Chengye Wang, Hanfang Sun, Shiro Yada et Azriel Rosenfeld. Some experiments in relaxation image matching using corner features. *Pattern Recognition*, 16(2) :167–182, 1983.

[WZ00] G. Wolberg et S. Zokai. Image registration for perspective deformation recovery. In F. A. Sadjadi, editor, *Proc. SPIE Vol. 4050, p. 259-270, Automatic Target Recognition X, Firooz A. Sadjadi ; Ed.*, pages 259–270, août 2000.

[XJ04] Quanren Xiong et Christopher Jaynes. Multi-resolution background modeling of dynamic scenes using weighted match filters. In *VSSN '04 : Proceedings of the ACM 2nd international workshop on Video surveillance & sensor networks*, pages 88–96, New York, NY, USA, 2004. ACM Press.

[Yeg99] A.F. Yegulalp. Minimum entropy sar autofocus. In *ASAP 99*, MIT Lincoln Laboratory, mars 1999.

[YH94] G. Yang et T. S. Huang. Human face detection in complex background. *Pattern Recognition*, 27(1) :53–63, 1994.

[ZC93] Qinfen Zheng et Rama Chellappa. A computational vision approach to image registration. *IEEE Transactions on Image Processing*, 2(3) :311–326, 1993.

[ZFS02] Barbara Zitová, Jan Flusser et Filip Sroubek. Application of image processing for the conservation of the medieval mosaic. In *Proceedings of the International Conference on Image Processing*, pages 993–996, 2002.

[ZK99] F. Zana et J.C. Klein. A multimodal registration algorithm of eye fundus images using vessels detection and hough transform. *IEEE Trans. Medical Imaging*, 18(5) :419–428, mai 1999.

[ZS03] J. Zhong et S. Sclaroff. Segmenting foreground objects from a dynamic textured background via a robust kalman filter. In *International Conference on Computer Vision*, octobre 2003.

Table des figures

1 Emplacements standard des caméras. (a) : caméra chauffeur, (b) : caméra couloir, (c) : caméra arrière 12

1.1 Schéma d'un système optique 19

2.1 Norme du gradient calculé par le filtre de Sobel et histogramme des normes 35
2.2 Détection des contours dans une image. (a) : image originale (b) : détection par le filtre de Canny (c) : détection par le filtre de Sobel 36
2.3 Carte des contours stables 37
2.4 Contribution d'un point de contour en fonction de la date de son apparition, pour une fenêtre temporelle de taille $T = 100$ 39
2.5 Masques de chanfrein 42
2.6 Carte de distances aux contours stables 43
2.7 Contours stables sur deux vues différentes de la même caméra 44
2.8 Superposition de la carte de distances et des contours stables d'une autre vue 44
2.9 Exemples de cartes de contours stables générées artificiellement 45
2.10 Distance estimée en fonction de la distance réelle obtenue sur les cartes de contours stables générées 46
2.11 Correction de la distortion radiale. à gauche : image originale. à droite : image corrigée 47
2.12 Adéquation du profil moyen du gradient avec le modèle de contour. Image nette. 50
2.13 Adéquation du profil moyen du gradient avec le modèle de contour. Image floue. 50
2.14 Corrélation entre les deux mesures de netteté proposées 52
2.15 Invariance au contenu des mesures de netteté pour une caméra correctement mise au point 53
2.16 Invariance au contenu des mesures de netteté pour une caméra légèrement défocalisée 53
2.17 Invariance au contenu des mesures de netteté pour une caméra fortement défocalisée 53
2.18 Différents type d'obstruction de la vue. (a) : opaque totale. (b) : opaque partielle. (c) : semi-transparente. (d) : texturée 54
2.19 Carte de contours stables d'une vue partiellement occultée 55

2.20 Carte de contours stables d'une vue occultée par un objet semi-transparent 56
2.21 Découpage en zones de l'image pour la mesure d'obstruction 57

3.1 Évolution de la mesure de décadrage (distance médiane entre contours stables de référence et courants calculée par la carte de distances aux contours stables courants) lors de l'apparition d'un décadrage de la caméra à l'image 3754 62
3.2 Évolution de la mesure d'obstruction du champ de vision (proportion de contours stables de référence visibles) lors de l'apparition d'un défaut . . . 63
3.3 Évolution des mesures de netteté lors de l'apparition d'un défaut de mise au point 64
3.4 Défauts présents dans les séquences de test 66
3.5 Évolution des mesures de netteté pendant un réglage de mise au point . . . 69
3.6 Évolution de la distance médiane d (en pixels) entre contours pendant un réglage de position, par rapport à la position de référence figure 3.7 71
3.7 (a) : Position de référence pour la simulation d'aide au positionnement. (b) : Image 12, position optimale de la caméra par rapport à la mesure de distance médiane (distance de 30 pixels) 72

4.1 Schéma global du détecteur de personnes 84

5.1 Répartition de la teinte chair dans l'espace couleur RGB 92
5.2 Répartition de la teinte chair dans l'espace de chrominance rg-normalisé . . 93
5.3 Échantillons de peau issus de la base d'images FERET 95
5.4 Détection de peau. (a) : Image originale (b) : Carte de probabilité de teinte chair 95
5.5 Photos d'une main avec des arrière-plans et éclairages de couleurs variées. Taux de bonnes détections/fausses détections 97
5.6 Photos d'une main compressées en JPEG avec différents facteurs de qualité Q 98
5.7 Performance de la détection de teinte chair en fonction du taux de compression JPEG 98
5.8 Profils temporels de pixels de la partie fixe et la partie vitrée 99
5.9 Dérivée de l'intensité des profils temporels de pixels de la partie fixe et la partie vitrée 100
5.10 Puissance moyenne de la dérivée du profil temporel de chaque pixel 101
5.11 Détections de portes (quadrilatères blancs) à partir des cartes de probabilité d'appartenance à une vitre 104
5.12 Soustraction entre images consécutives. (a) et (b) : Images originales. (c) : Résultat de la soustraction 106
5.13 Moyenne temporelle des images d'une séquence vidéo sur deux jours 106
5.14 (a) : image originale. (b) : soustraction entre image moyenne et image originale 107
5.15 Histogrammes de couleurs en RG-normalisé des valeurs prises par différents pixels au cours du temps 109

5.16 Détection des pixels d'intérêt avec le modèle gaussien d'arrière-plan 110
5.17 Inverse de la fonction de répartition de la distribution χ^2_4 119
5.18 Inverse de la fonction caractéristique de la distribution χ^2_{10} 123
5.19 Estimation de l'arrière-plan par la méthode présentée. (a) : Image originale. (b) :Arrière-plan estimé 126
5.20 Estimation de l'arrière-plan par filtrage médian temporel 127
5.21 Erreur quadratique moyenne entre arrière-plan réel et estimé. Comparaison avec l'estimation par image médiane 127
5.22 Détection de pixels d'avant-plan sur la séquence synthétique 129
5.23 Performance en bonnes détections et fausses détections de l'estimation de fond proposée sur la séquence synthétique 130
5.24 Performance en bonnes détections et fausses détections de l'estimation de fond par mélange de Gaussiennes sur la séquence synthétique 131
5.25 Détection de pixels d'avant-plan sur scène réelle 131

6.1 Modèle d'ellipse de peau 138
6.2 Détection de visage par le modèle d'ellipse de teinte chair 145
6.3 Résultats de détection sur séquences réelles avec le modèle d'ellipse. Les ellipses blanches correspondent aux visages réellement renvoyés par l'algorithme. Les autres ellipses sont des hypothèses avancées puis rejetées par l'algorithme 146
6.4 Exemple d'apparence de personnes en plan américain 148
6.5 Modèle de personne. A : région de la tête, B : région du visage, C : région du corps 148
6.6 Calcul rapide de la probabilité moyenne d'un rectangle à partir de l'image intégrale 152
6.7 Probabilités associées aux vecteurs $\boldsymbol{v}_{max}$ pendant l'étape de détection, en fonction de l'itération 154
6.8 Détection de personnes. (a) : Probabilités de peau. (b) : Probabilités d'avant-plan. (c) : Régions du visage et probabilités associées 156
6.9 Fausse détection de main levée. (a) : Probabilités de peau (b) : Probabilité d'avant-plan. (b) : Résultat de la détection de personne 157
6.10 Suivi de visages par prédiction de mouvement 161
6.11 Signature de couleurs obtenue par segmentation k-means 163

7.1 Schéma de la compression sélective par préfiltrage des images 169
7.2 Exemple d'image et cartes de probabilités associées pour la compression sélective. (a) : image originale. (b) : teinte chair. (c) : extérieur 171
7.3 Exemple de résultat du préfiltrage DCT. (a) :image originale, (b) : image préfiltrée 174
7.4 Exemple de résultat du préfiltrage gaussien 174
7.5 PSNR en fonction de la taille des données JPEG, pour les préfiltrages DCT et gaussien 175

7.6 Comparaison subjective des deux préfiltrages pour une même taille de fichier JPEG de 30ko. (a) : DCT. (b) : gaussien 176
7.7 Séquence vidéo de test pour la compression sélective. Peau détectée et résultat des deux préfiltrages 178
7.8 Gain en taille de la vidéo compressée par h264 par rapport à une compression sans préfiltrage, en fonction du facteur de lissage f 179
7.9 Comptage des visages passant un segment 180
7.10 Comptage de personne sur une séquence d'arrêt 182

Liste des tableaux

5.1 Comparaison des taux de détection entre l'estimation de fond proposée et l'estimation de fond par mélanges de Gaussiennes 128

6.1 Performance de la détection de visage avec le modèle d'ellipse sur la base Caltech 145
6.2 Taux de détection de visages avec le modèle d'ellipse de peau, sur 521 visages 146
6.3 Résultats de détection des personnes à un arrêt avec le modèle de personne complet 155
6.4 Comparaison des performances en détection de personnes entre le modèle d'ellipse de peau et le modèle de personne complet 165

7.1 Taille des données vidéo compressées par h264, avec et sans préfiltrage . . 177