HAL
open science

Détection de rails, caractérisation de croisements et localisation de trains sur la trajectoire d'un métro automatique

Jorge Corsino Espino

To cite this version:

Jorge Corsino Espino. Détection de rails, caractérisation de croisements et localisation de trains sur la trajectoire d'un métro automatique. Autre [cs.OH]. Ecole Nationale Supérieure des Mines de Paris, 2014. Français. NNT : 2014ENMP0013 . pastel-01068899

HAL Id: pastel-01068899

https://pastel.hal.science/pastel-01068899v1

Submitted on 26 Sep 2014

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.

ParisTech
INSTITUT DES SCIENCES ET TECHNOLOGIES
PARIS INSTITUTE OF TECHNOLOGY

MINES ParisTech

École doctorale n°432 : Sciences des Métiers de l'ingénieur

Doctorat ParisTech

THÈSE

pour obtenir le grade de docteur délivré par

l'École nationale supérieure des mines de Paris

Spécialité « Informatique temps-réel, robotique et automatique »

présentée et soutenue publiquement par

Jorge CORSINO ESPINO

le 13 Juin 2014

Détection de rails, caractérisation de croisements et localisation de trains sur la trajectoire d'un métro automatique

Rails detection, turnouts characterisation and trains localization in an automated metro's trajectory

Directrice de thèse : **Brigitte d'ANDRÉA-NOVEL**
Co-encadrement de la thèse : **Bogdan STANCIULESCU**
Co-encadrement de la thèse : **Philippe FORIN**

Jury

M. Didier AUBERT, Directeur de Recherche, IFSTTAR	Rapporteur
M. Antoine MANZANERA, Maître de Conférence (HDR), ENSTA ParisTech	Rapporteur
Mme. Brigitte d'ANDRÉA-NOVEL, Professeur, Mines ParisTech	Examinateur
Mme. Bernadette DORIZZI, Directrice de Recherche, TELECOM SudParis	Examinateur
M. Philippe FORIN, Ingénieur, SIEMENES SAS	Examinateur
M. Bogdan STANCIULESCU, Maître de Conférence, Mines ParisTech	Examinateur
M. Said EL FASSI, Ingénieur, SNCF	Invité

THÈSE

MINES ParisTech
Centre de Robotique
60 bd Saint Michel, 75006 PARIS - France

Remerciements

Je remercie Monsieur Didier Aubert et Monsieur Antoine Manzanera qui m'ont fait l'honneur d'accepter d'être rapporteurs de cette thèse. La version finale de ce mémoire a bénéficié de leur lecture très attentive et de leurs remarques précieuses.

Mes remerciements s'adressent aussi à Madame Bernadette Dorizzi pour avoir accepté d'être président du jury. Je remercie également tous les membres du jury d'avoir accepté d'assister à la présentation de ce travail, particulièrement à Monsieur Said EL Fassi, promoteur de cette thèse CIFRE.

Je voudrais remercier Madame Brigitte d'Andréa-Novel d'avoir dirigé les travaux de thèse et de m'avoir accueilli au sein du cycle d'Ingénieur Civil option MAREVA. Les connaissances acquises pendant cet année m'ont permit de mieux affronter les travaux de thèse.

Je remercie Monsieur Bogdan Stanciulescu qui a déposé sa confiance en moi pour réaliser une thèse et pour avoir encadré mes travaux. Il m'a toujours encouragé pour rédiger des articles et nous avons passés des très bons moments après des journées de travail intenses.

Je remercie également Monsieur Philippe Forin, co-encadrant de thèse côté industriel. Il a toujours été disponible, à l'écoute de mes nombreuses questions, et s'est toujours intéressé à l'avancée de mes travaux. Les nombreuses discussions que nous avons eues ainsi que ses conseils sont pour beaucoup dans le résultat final de ce travail.

Un remerciement spécial pour les personnes qui ont accepté de relire quelques parties du manuscrit : Martyna, Anne Sophie, Fatin, Julien, Benjamin, João, Olivier, Sylvain, Omar,...

Puis, je souhaite vivement remercier tous mes collègues du CAOR et de SIEMENS. Ces trois années de bonne ambiance et de collaboration seront inoubliables. Certains d'entre vous sont devenus des amis.

Enfin sans avoir directement participé au travail de recherche, je remercie toutes les personnes qui ont aidé à son accomplissement et qui m'ont accompagné à Siemens et à l'Ecole de Mines de Paris.

Je dois beaucoup à ma famille et mes amis. Je ne les citerai pas, je pense simplement à eux.

Table des matières

1 Introduction **1**
1.1 Introduction . . . 1
1.2 Contexte . . . 3
1.2.1 Sûreté ferroviaire . . . 3
1.2.2 Normes EN 50129 et EN 50126 . . . 4
1.2.3 Norme EN 15227 . . . 6
1.3 La vitesse commerciale . . . 7
1.3.1 Marche du train . . . 7
1.3.2 Distance d'arrêt . . . 8
1.4 Conduites non automatisées du train . . . 9
1.4.1 Ouverture de la ligne . . . 9
1.4.2 Reprise en conduite manuelle . . . 10
1.4.3 Mouvement en garage ou atelier . . . 11
1.5 Objectifs . . . 12
1.5.1 Objectifs techniques et performances à atteindre . . . 12
1.5.2 Objectifs scientifiques . . . 12
1.6 Organisation du manuscrit . . . 14

2 La plateforme mobile **15**
2.1 État de l'art des systèmes de détection d'obstacles ferroviaires . . . 15
2.2 Choix du capteur . . . 19
2.2.1 Capteurs actifs . . . 19
2.2.2 Capteurs passifs . . . 20
2.3 Le Véhicule Automatique Léger : VAL . . . 21
2.3.1 Composition d'un véhicule . . . 21
2.3.2 Caractéristiques . . . 22
2.3.3 Modèle du véhicule . . . 23
2.4 Étalonnage . . . 24
2.4.1 Modéliser une caméra . . . 24
2.4.2 Méthodes d'étalonnage . . . 27
2.4.3 Méthode d'étalonnage proposée . . . 32
2.4.4 Résultats . . . 35

3 Détection de la voie **39**
3.1 Motivations . . . 39
3.1.1 Base de données . . . 39
3.1.2 La piste de roulement . . . 40
3.2 État de l'art et approches testées . . . 41
3.3 Approche retenue . . . 51
3.3.1 Sélection du Masque . . . 51
3.3.2 Analyse des gradients . . . 53
3.3.3 Recherche du modèle dans l'espace 3D . . . 57
3.3.4 Dépendance temporelle . . . 62
3.3.5 Détermination de la distance maximale de détection . . . 63
3.4 Résultats . . . 67

4 Détection et caractérisation de croisements **73**
4.1 Motivations . . . 73
4.2 Les appareils de voie . . . 75
4.3 Méthode proposée pour la détection des appareils de voie . . . 77
4.3.1 Utilisation de la transformée *Bird eyes' view* . . . 77
4.3.2 Descripteur : *Template matching* . . . 78
4.3.3 Descripteur : Histogramme de gradients orientés (HOG) . . . 81
4.4 Résultats de détection . . . 84
4.4.1 Machines à vecteurs de support (SVM) . . . 84
4.4.2 Matrice de confusion et courbe ROC . . . 85
4.4.3 Résultats du descripteur basé en *Template matching* . . . 86
4.4.4 Résultats obtenus avec HOG . . . 87
4.5 Méthode proposée pour la classification du type d'appareil de voie . . . 89
4.5.1 SVM multi-classe . . . 90
4.5.2 Descripteur proposé : HOG . . . 90
4.5.3 Résultats de classification . . . 92

5 Détection d'obstacles : trains **97**
5.1 Introduction . . . 97
5.1.1 Systèmes en commercialisation industrielle . . . 99
5.2 Base de données . . . 100
5.3 Sélection du descripteur . . . 102
5.3.1 Descripteurs locaux . . . 102
5.3.2 Analyse globale : Histogramme de gradients orientés (HOG) . . . 106
5.4 Résultats . . . 107

6 Conclusions **111**
6.1 Conclusions . . . 111
6.2 Travaux futurs . . . 113

Publications **115**
A.1 Article ITSC 2012 . . . 116
A.2 Article ICIRT 2013 . . . 121
A.3 Article ITSC 2013 . . . 127

Bibliographie **133**

Chapitre 1

Introduction

Sommaire

1.1 Introduction **1**
1.2 Contexte **3**
1.2.1 Sûreté ferroviaire 3
1.2.2 Normes EN 50129 et EN 50126 4
1.2.2.1 Niveau de probabilité de l'accident potentiel 4
1.2.2.2 Niveau de gravité des dommages engendrés par l'accident potentiel 5
1.2.2.3 Niveau de risque 5
1.2.3 Norme EN 15227 6
1.3 La vitesse commerciale **7**
1.3.1 Marche du train 7
1.3.2 Distance d'arrêt 8
1.4 Conduites non automatisées du train **9**
1.4.1 Ouverture de la ligne 9
1.4.2 Reprise en conduite manuelle 10
1.4.2.1 Actions du système 10
1.4.3 Mouvement en garage ou atelier 11
1.4.3.1 Actions du système 11
1.5 Objectifs **12**
1.5.1 Objectifs techniques et performances à atteindre 12
1.5.2 Objectifs scientifiques 12
1.6 Organisation du manuscrit **14**

1.1 Introduction

Dans les dernières décennies, les systèmes de transports ont vu s'améliorer les conditions de sécurité, l'optimisation des réseaux, la réduction de la consommation d'énergie et la diminution de la pollution. Les recherches autour de la conduite automatique, dans laquelle de nouvelles techniques sont en train de se développer, se poursuivent. Ces recherches incluent la localisation et le suivi des marquages des voies de circulation en maintenant la distance de sécurité par rapport aux autres véhicules, les changements de voie pour dépasser un véhicule, la détection d'obstacles, la gestion des carrefours et l'amélioration des conditions de visibilité des caméras. Ils visent au développement de Systèmes Intelligents de Transports (STI) ou «*Intelligent Transport Systems*» (ITS).

Les premières recherches en matière de «véhicule intelligent» et de «route intelligente» ont été réalisées au milieu des années 1980 avec le projet Européen PROMETHEUS (Acronyme de *PROgraM for an European Traffic with Highest Efficiency and Unprecedent Safety*) en Europe, aux États-Unis le projet IVHS (*Intelligent Vehicle and Highway System*) et au Japon VICS *(Vehicle Information and Communication System)* [Massimo *et al.* 2000].

Les transports en commun jouent un rôle essentiel dans les actions de développement durable et le métro s'est imposé dans toutes les grandes métropoles comme la solution la plus efficace en matière de transport urbain de masse, car il présente de nombreux avantages par rapport aux véhicules personnels comme la rapidité, la sécurité, la fiabilité, une faible pollution et un coût suffisamment bas pour être accessible à tous.

Actuellement, les systèmes de transport urbains entièrement automatiques se développent, car ils présentent de nombreux avantages pour les exploitants. L'automatisation permet plus de sécurité et une meilleure adaptation aux besoins du trafic, tout en offrant une grande attractivité pour les passagers.

Siemens S.A.S. a créé et développé le premier métro au monde entièrement automatique et sans conducteur en 1983 à Lille sous le nom de VAL (*Véhicule Automatique Léger*). D'abord réservée aux métros légers et économiques, l'automatisation a depuis été appliquée à des métros de gabarit plus important comme METEOR (MÉTro Est-Ouest Rapide à Paris ligne 4). Sa souplesse d'exploitation le rend également parfaitement adapté aux dessertes d'aéroports. De ce fait, il est déjà en exploitation à Paris Orly, Paris Charles de Gaulle et Chicago [Provost 2005].

L'amélioration de la sûreté et la sécurité sont primordiales pour n'importe quel système de transport et ont entraîné ces dernières années une diminution du nombre d'accidents. Une analyse des accidents ferroviaires grâce à des données extraites de l'Eurostat (centre européen chargé de l'information statistique à l'échelle communautaire) [NCAP 2013], constate que les deux principales causes d'accidents sont les passages à niveau et les collisions (tableau 1.1).

Tableau 1.1 Accidents ferroviaires - nombre de victimes par type d'accident, UE-27, 2012

	2008		2009		2010		2011	
	blessés	décès	blessés	décès	blessés	décès	blessés	décès
Collision	92	23	53	10	267	41	49	15
Déraillement	15	3	19	32	21	3	45	4
Passages à niveau	453	399	354	433	381	378	329	317

Bien que les accidents ferroviaires soient peu nombreux par rapport aux accidents routiers, leur impact médiatique et les perturbations de circulation conduisent les exploitants et les autorités de tutelle à améliorer en permanence la sécurité des circulations.

Pour définir une solution industrielle pour les métros automatiques, nous souhaitons évaluer les approches basées sur la vision par ordinateur et le traitement automatisé des images, avec l'objectif de proposer des mesures préventives contre les collisions.

Cette thèse porte sur la fonction de détection d'obstacles dans le domaine ferroviaire à partir de la vision par ordinateur. Il s'agit d'assurer une perception de l'environnement situé à l'avant du train par l'intermédiaire d'un capteur optique afin de détecter et d'évaluer les distances et tailles des obstacles situés sur la voie.

La liste suivante est une première identification des applications ferroviaires pour lesquelles des systèmes de traitements d'images existent (mais qui posent parfois encore des problèmes de mise au point) :

- Surveillance de passage à niveau (voiture ou camion arrêtés, personnes) : caméras fixes.
- Surveillance de périmètre (observation des intrusions dans des lieux interdits comme les tunnels, les ponts et la voie) : caméras fixes.
- Détection de chute d'objet ou de personne sur la voie en station : caméras fixes.
- Détection d'objet abandonné sur un quai ou dans un couloir : caméras fixes.
- Surveillance des infrastructures et de la caténaire : caméra montée sur le train.
- Surveillance dans les véhicules : enregistrement des données pour enquête après un incident, détection de comportements anormaux (mise au point difficile).

Ces solutions sont souvent installées dans un environnement contrôlé. La mise en œuvre en devient simplifiée par la définition d'un ensemble de paramètres permettant d'atteindre les objectifs opérationnels. Cette méthodologie est difficilement applicable dans notre contexte, dans lequel les variations de luminosité sont de grande amplitude et l'environnement scénique évolue avec une grande rapidité. Les objectifs de sécurité opérationnelle des fonctions doivent être compatibles avec le niveau de sécurité des automatismes de conduite.

1.2 Contexte

De nombreux projets de recherche ont été menés ou sont en cours pour faire avancer les développements matériels et logiciels vers l'automatisation totale de la conduite et pour l'amélioration des conditions de sécurité et l'optimisation des transports.

Nous cherchons à mettre au service du métro automatique les avancées dans le domaine du traitement d'image développées pour les ITS. Dans le cas de voitures intelligentes, il existe de nombreuses études de systèmes d'aide à la conduite utilisant des caméras. Par contre, les caméras sont aujourd'hui très peu présentes dans les trains. La liste suivante est une première identification des applications ferroviaires pour lesquelles des systèmes de traitements d'images existent : surveillance de passage à niveau [Vazquez *et al.* 2004] (voiture ou camion arrêtés, personnes) avec des caméras fixes pour analyser la scène ; surveillance de périmètre (observation des intrusions dans des lieux interdits : tunnels, ponts et voie) [Garcia *et al.* 2005] ; détection de chute d'objets ou de personnes sur la voie en station [Kawamura *et al.* 2011] ; surveillance des infrastructures et des caténaires avec une caméra montée sur le train [Puhlmann *et al.* 2004].

Avec l'implantation des caméras dans la locomotive des métros automatiques, nous cherchons à contribuer à l'amélioration de la sécurité et de la sûreté des transports, et par conséquent à augmenter la productivité et les services dans les transports publics [Nangtin *et al.* 2005].

1.2.1 Sûreté ferroviaire

Le métro est un transport en commun en site propre (TCSP), c'est-à-dire, un transport qui emprunte une voie ou un espace qui lui est réservé. Le métro entièrement automatique est considéré lui comme un «TCSP intégral» dans la mesure où il circule généralement en souterrain ou en viaduc et que son chemin n'est jamais traversé par une autre route.

Ainsi, les principaux risques pour ces systèmes ferroviaires sont limités : déraillement, collision de trains par rattrapage, prise en écharpe dans les appareils de voie (essentiellement les aiguillages) et choc frontal.

La sûreté de fonctionnement des systèmes industriels est définie par l'absence de risque inacceptable, de blessure ou d'atteinte à la santé des personnes, directement ou indirectement, résultant d'un dommage dû au matériel ou à l'environnement. Pour ces systèmes, les pannes peuvent provenir de nombreux facteurs différents :

- Erreurs logicielles ;
- Erreurs humaines ;
- Influence de l'environnement ;
- Panne matérielle des mécanismes ;

Le terme sécurité peut avoir deux sens, le premier (sécurité-confidentialité) a trait à une volonté humaine de causer une défaillance, tandis que le second sens (sécurité-innocuité) est l'absence de conséquences catastrophiques pour les usagers et l'environnement. C'est essentiellement cette seconde notion qui nous intéresse pour le développement d'applications dans les transports automatiques.

1.2.2 Normes EN 50129 et EN 50126

C'est pourquoi la norme européenne 50129 fut créée [Européene 2003a]. Elle définit le niveau de sécurité intégré de l'anglais (SIL-*Security Integrity Level*). La norme détaille les exigences pour atteindre chaque niveau d'intégrité de la sécurité. Ces exigences sont plus sévères aux niveaux d'intégrité de la sécurité les plus élevés, de manière à garantir une probabilité de défaillance dangereuse plus basse. Ces exigences de qualité constituent les critères de fiabilité du système en fonction des risques.

Le niveau SIL sert à apprécier la probabilité de défaillance dangereuse par heure (PFH-*Probability Failure per Hour*) et le Taux maximum acceptable d'occurrence d'un danger (THR) par heure des fonctions de sécurité des systèmes électriques et électroniques programmables et est défini selon 4 niveaux de sécurité (tableau 1.2). Chaque niveau correspond à une plage de probabilité de défaillance d'une fonction de sécurité. Plus le niveau d'intégrité de sécurité des systèmes de sécurité est élevé, plus il est probable qu'ils exécutent efficacement les fonctions de sécurité demandées. Le SIL 4 est le système de sécurité le plus élevé et le SIL 1 le moins élevé.

Tableau 1.2 Table des niveaux SIL

Niveau de Sécurité Intégré (SIL)	Taux maximum acceptable d'occurrence d'un danger (THR) par heure et par fonction
SIL 4	THR compris entre 10^{-9} et 10^{-8}
SIL 3	THR compris entre 10^{-8} et 10^{-7}
SIL 2	THR compris entre 10^{-7} et 10^{-6}
SIL 1	THR compris entre 10^{-6} et 10^{-5}

L'automatisation intégrale des trains est soumise à un niveau de sécurité SIL 4 et son rôle principal est d'assurer :
- l'autonomie et la sécurité du mouvement des trains,
- le respect de la signalisation,
- la localisation,
- la transmission d'informations,
- la régulation de vitesse.

La norme européenne EN 50129 définit la sécurité par « l'absence de tout niveau de risque inacceptable ». De même, EN 50126 [Européene 2003b] définit le niveau de gravité en associant les quantifications avec les conséquences engendrées par l'accident potentiel. Pour définir et évaluer le niveau d'acceptabilité des risques, cette norme propose la matrice gravité/occurrence [Mabrouk 2010].

Nous allons détailler les définitions du niveau de probabilité, du niveau de gravité et du niveau de risque, pour pouvoir comprendre le choix des techniques de traitement d'images proposées pour satisfaire aux exigences ferroviaires.

1.2.2.1 Niveau de probabilité de l'accident potentiel

La norme EN 50126 propose des probabilités quantitatives, d'où la probabilité d'occurrence d'un événement peut être :

- **Fréquent** : surviendra probablement souvent. Le risque de concrétisation du danger sera continuellement présent ($P > 10^{-3}/h$).
- **Probable** : surviendra plusieurs fois. Le danger se concrétisera fréquemment ($10^{-3}/h > P > 10^{-4}/h$).
- **Occasionnel** : surviendra probablement plusieurs fois au cours de la vie de système. Le danger se concrétisera plusieurs fois ($10^{-4}/h > P > 10^{-5}/h$).

- **Rare** : surviendra probablement au cours de la vie du système. On peut raisonnablement s'attendre à la concrétisation de ce danger ($10^{-5}/h > P > 10^{-7}/h$).
- **Improbable** : peu probable mais possible. On peut admettre que ce danger se concrétisera exceptionnellement ($10^{-7}/h > P > 10^{-8}/h$).
- **Invraisemblable** : Extrêmement improbable. On peut admettre que ce danger ne se concrétisera pas. ($10^{-9}/h > P$).

Pour notre cas, la probabilité horaire P s'applique à la ligne de métro complète (tous les équipements de la conduite automatique fixes et embarqués).

1.2.2.2 Niveau de gravité des dommages engendrés par l'accident potentiel

Les niveaux de gravité des dommages engendrés par l'accident potentiel peuvent être analysés sur trois aspects : selon les dommages aux personnes, au système et à l'environnement. Considérant les conséquences sur les personnes, la norme EN 50126 propose quatre niveaux de gravité des dommages :

- **Catastrophique** : plusieurs blessés graves ou plusieurs morts.
- **Grave** : un blessé grave ou un mort.
- **Signifiant** : un blessé léger.
- **Insignifiant** : ni blessé, ni mort.

1.2.2.3 Niveau de risque

Généralement, il existe une confusion entre le risque et le niveau de risque. En effet, le niveau de risque identifie la combinaison du niveau de probabilité d'occurrence de l'accident potentiel ainsi que le niveau de gravité des dommages engendrés par cet accident potentiel ; d'où plusieurs classifications apparaissent. La norme EN 50126 identifie 4 niveaux de risque :

- **Risques intolérables** : doivent être éliminés.
- **Risques non souhaitables** : ne peuvent être acceptés, avec l'accord de l'Exploitant, que si l'on ne peut pas réduire le risque.
- **Risques tolérables** : acceptables, avec l'accord de l'Exploitant, et moyennant des précautions appropriées.
- **Risques négligeables** : acceptables, avec l'accord de l'Exploitant.

Finalement, à partir de ces définitions, les différentes normes recommandent la mise en place d'une matrice gravité/occurrence pour évaluer le risque. Un exemple de cette matrice est montré dans la figure 1.1.

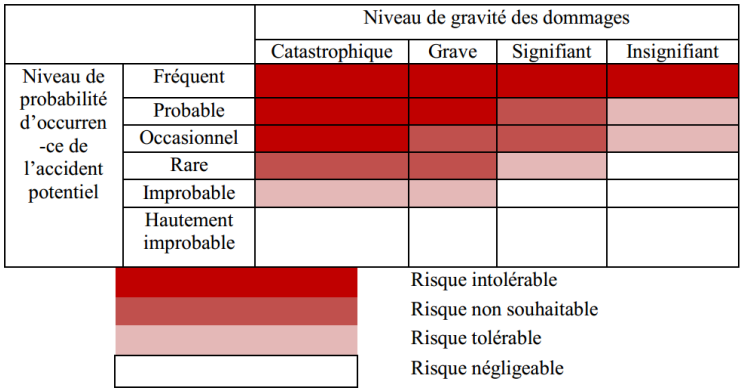


FIGURE 1.1 – Matrice gravité/occurrence. Image extraite de [Mabrouk 2010].

1.2.3 Norme EN 15227

Applicable depuis le 1er janvier 2011, la norme EN15227, relative à la sécurité passive en cas de collision frontale, s'impose à tous les nouveaux véhicules ferroviaires. Appliquée aux métros, elle stipule que «l'espace de survie» (espace normalement occupé par les passagers ou le conducteur) soit préservé en cas de collision frontale d'une rame roulant à 25km/h contre une rame arrêtée. Cependant si les conditions d'exploitation sont telles qu'un scénario de collision ne peut pas se produire, ou s'il existe des preuves montrant que la probabilité pour que ce scénario se produise est si faible qu'elle est largement acceptable, alors, il n'est pas nécessaire de le considérer pour la conception du véhicule.

C'est précisément le cas pour le système VAL et, plus généralement, pour les métros en conduite automatique intégrale (systèmes «manless») **à l'exception des phases de reprise en conduite manuelle suite à une indisponibilité du «pilote automatique»**. L'anticollision repose alors entièrement sur le conducteur, avec les risques inhérents à ce mode de conduite (malaise, perte d'attention,...). Ces phases étant néanmoins rares et de courtes durées, l'ajout d'un dispositif anti-collisions opérationnel pendant la conduite manuelle à vue permettrait de satisfaire aux exigences de la norme.

De plus, la conduite manuelle à vue est extrêmement perturbante pour la disponibilité de la ligne. Sa vitesse commerciale se voit rapidement perturbée. L'agent de conduite n'étant pas à bord, il est nécessaire de distribuer le long de la ligne du personnel pour réduire le temps d'intervention. Pour limiter les impacts du temps d'intervention sur la disponibilité de la ligne, **il semble intéressant de développer les moyens permettant d'augmenter la vitesse de conduite à vue, tout en respectant la norme EN 15227**.

Pour la détections des petits obstacles, actuellement, des mesures de protection de l'environnement du véhicule sont mises en œuvre pour éviter toute intrusion d'objet dans le domaine. Au cas où, malgré tout, un objet serait sur le plan de roulement du véhicule, une traverse «chasse corps» est montée à l'avant des rames VAL, juste au-dessus du plan de roulement. Elle a pour rôles d'une part de chasser les éventuels corps étrangers (cailloux, débris...) présents sur les pistes de roulement pour ne pas endommager les pneus et d'autre part de déclencher le freinage d'urgence pour arrêter le train au plus tôt après un choc.

Dans ce cas, un agent doit être envoyé sur place pour décider si le train peut ou non repartir. Or, dans la pratique, la plupart des déclenchements sont dus à des causes qui ne justifient pas l'envoi d'un agent sur place (choc avec un lapin sur OrlyVal par exemple). Une caméra placée derrière le pare-brise, associée à un enregistreur et à une transmission vers le PCC permettrait de faire repartir le train au plus tôt après un choc bénin. La figure 1.2 nous montre un exemple de cette barre.

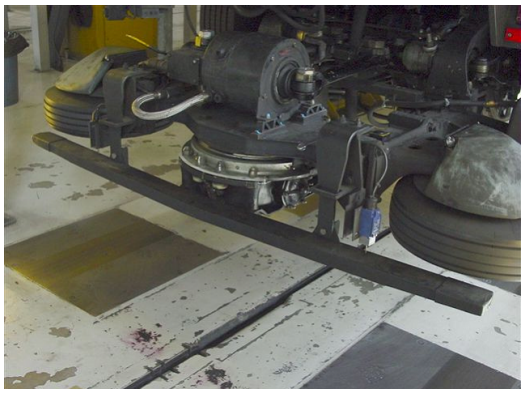


FIGURE 1.2 – La traverse chasse-corps du VAL.

Compte tenu de la faible proportion de temps durant laquelle des collisions sont possibles, l'adjonction à la conduite manuelle, d'un dispositif de détection automatique d'obstacles à partir du traitement d'images, permettrait de satisfaire aux exigences de sécurité de cette norme sans procéder à des modifications structurelles des rames.

De cette manière, l'ajout de la détection d'obstacles et de la visualisation des voies permettrait, de plus, selon les cas :

- La détection de fausses alarmes dues aux petits obstacles.
- Les mouvements automatiques de trains en garage, sans équiper tout le faisceau de voies avec les dispositifs de localisation et de signalisation.

1.3 La vitesse commerciale

La vitesse commerciale est un des facteurs qui détermine la qualité des prestations des transports publics et leur attractivité. Elle représente le rapport entre la longueur du trajet et le temps nécessaire pour le parcourir, y compris la durée des arrêts intermédiaires. Elle comprend, la durée du parcours entre les stations, le temps d'attente aux stations, le temps d'attente aux feux de circulation, ainsi que les autres temps d'arrêt (incidents, accidents, etc.).

Les principaux facteurs qui influencent la vitesse commerciale sont :

- La vitesse maximale que peuvent atteindre les véhicules, qui dépend de leur motorisation, mais aussi de l'infrastructure qu'ils vont utiliser : site propre exclusif, entièrement protégé ; site propre partagé ; voirie banalisée.
- La longueur des inter-stations, qui détermine le nombre d'arrêts sur la ligne.
- Les puissances d'accélération et de freinage, qui jouent un rôle d'autant plus important que le nombre d'arrêts est élevé.
- Le temps d'arrêt en station, qui dépend en moyenne du nombre de voyageurs qui montent et qui descendent, mais aussi de l'organisation des portes et des espaces intérieurs des véhicules.

Des exemples de vitesses commerciales des métros sont : 21 km/h pour la ligne 4 de Paris et de 39 km/h pour la ligne 14 de Paris. Les lignes de métro automatique léger (VAL) ont une vitesse commerciale comprise entre 30 et 35 km/h. Ceci est une conséquence de la meilleure optimisation de certains paramètres pour les métros fonctionnant sur des infrastructures qui leur sont totalement dédiées comme les temps d'arrêt en station et les capacités d'accélération et de freinage.

1.3.1 Marche du train

Pour les transports ferroviaires, la marche type (relation entre la vitesse et la position) d'un train entre deux stations où il s'arrête peut être découpée en trois phases :

- Phase 1 d'accélération entre la vitesse initiale nulle et la vitesse maximale.
- Phase 2 de marche à vitesse constante.
- Phase 3 de freinage entre la vitesse maximale et la vitesse finale nulle.

Lorsque les tronçons sont courts, il peut arriver que le convoi n'ait pas la possibilité d'atteindre la vitesse maximale ; dans ce cas, la phase d'accélération est immédiatement suivie d'une phase de freinage.

Les deux cas sont illustrés dans la figure 1.3, indiquant la vitesse en fonction de la position du train.

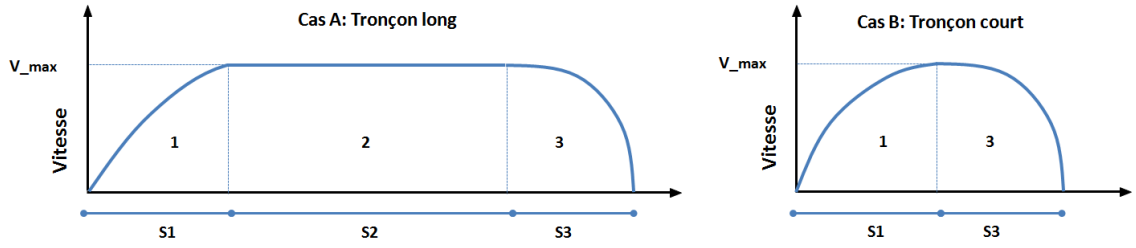


FIGURE 1.3 – Courbes de vitesse du train en fonction de la taille du tronçon.

$$s_1 \approx \frac{V^2}{2a}; \quad t_1 = \frac{V}{a}$$
$$s_2 = V \cdot t_2 \tag{1.1}$$
$$s_3 \approx \frac{V^2}{2d} \quad t_3 = \frac{V}{d}$$

En autre, en circulation ferroviaire, nous pouvons différencier 4 types de marche :
- **La marche normale** : la vitesse limite du train dépend de la section de ligne concernée, de la vitesse limite des différents éléments du train ainsi que de sa capacité de freinage.
- L'obligation de **marcher en manœuvre** impose au conducteur de s'avancer avec prudence, sans dépasser la vitesse de 30 km/h et en se tenant prêt à obéir aux signaux qu'il pourrait rencontrer et d'obéir à toute injonction d'un agent de manœuvre.
- **La marche à vue** : elle impose à un conducteur de s'avancer avec prudence, compte tenu de la partie de voie qu'il aperçoit devant lui, de manière à pouvoir s'arrêter avant une queue de train, un signal d'arrêt ou un obstacle. Elle est généralement limitée à 18km/h pour le VAL.
- **La marche avec prudence** impose à un conducteur de limiter sa vitesse compte tenu du motif qui lui a été indiqué.

1.3.2 Distance d'arrêt

Le code de la route définit la distance d'arrêt comme suit : «*La distance d'arrêt d'un véhicule correspond à la distance parcourue pendant le temps de réaction de son conducteur et la distance de freinage*» ; le temps de réaction du conducteur en cas d'incident est évalué typiquement à 1 seconde.

Ainsi, la distance $d_{parcourue}$ est la somme de la distance de freinage et de la distance parcourue par le train pendant la mise en route du freinage (c'est-à-dire avant que le freinage ne soit effectif). L'expression de la distance parcourue est donnée par l'équation 1.2, où le temps de réaction est $T_{système}$, la vitesse du train est v, sa décélération γ (plus γ est grand, meilleur est le freinage), la vitesse à attendre v_{but}, g est l'accélération de la pesanteur et i représente la déclivité du terrain pour mille.

$$D_{parcourue} = t_{système} v + \frac{v^2 - v_{but}}{2(\gamma - g \cdot i)} \tag{1.2}$$

Si nous considérons que le temps réaction du système complet (dispositif de détection d'obstacles + transmission + chaîne de freinage) dans le pire cas est égal à $T_{système} = 1$ s, une déclivité nulle et l'arrêt complet du train $v_{but} = 0$, la distance d'arrêt se réduit à l'équation 1.3

$$D_{arrêt} = v + \frac{v^2}{2\gamma} \tag{1.3}$$

Nous proposons d'analyser la distance d'arrêt pour différentes vitesses de notre train VAL, sachant que la décélération minimale assurée en cas de freinage d'urgence est égale à $\gamma = 1.8m/s^2$. Pour la vitesse maximale d'exploitation (*80km/h*) la distance d'arrêt est de *160 m*. Dans le cas d'une conduite manuelle en mode dégradé où la vitesse est limitée à *18km/h* nous avons une distance d'arrêt de *12 m*.

1.4 Conduites non automatisées du train

L'objectif du projet est d'ameliorer la sécurité des personnes et des équipements des métros automatiques sans conducteur à partir d'une approche basée sur la vision par ordinateur. En particulier, l'étude porte sur un dispositif permettant de satisfaire aux exigences de la norme EN 15227 sans modifier la structure de la caisse existante **tout en augmentant si possible la vitesse maximale de la marche à vue**. L'étude porte sur la fonction détection d'obstacle, en particulier, la prévention des collisions de trains.

La première solution qui vient à l'esprit pour réaliser un dispositif anticollisions basé sur la vision consiste à rechercher les obstacles potentiels devant le train et à les confronter à une base de données des obstacles possibles (dans notre cas, une autre rame arrêtée). Cependant, compte-tenu du haut degré de sécurité demandé dans le monde ferroviaire, assurer la détection au seuil de probabilité requis (>0,999) conduirait à un taux de faux positifs inacceptable en raison des perturbations engendrées sur la ligne. De plus, ces perturbations elles-mêmes étant des sources potentielles d'accidents, la sécurité globale n'en serait pas améliorée, voire serait dégradée.

Ces raisons nous obligent à concevoir le système de manière différente. Nous proposons d'utiliser la détection de la voie pour définir une zone libre d'obstacles (trains). Cette détection de la voie n'est basée que sur la reconnaissance des rails dans l'image et permet de déterminer la distance maximale de reconnaissance de la voie. Au-delà de cette distance, il existe trois hypothèses pour le système :

- La voie est libre mais elle n'est pas visible à cause de la résolution de la caméra, des conditions météorologiques, d'une flaque d'eau, des parois du tunnel en virage...
- La voie est occupée par un train.
- On se trouve à une des extrémités de la ligne et la voie arrive à sa fin.

Nous proposons de développer un système capable quelles que soient les conditions d'éclairage ou à la présence d'obstacles «normaux» (les parois du tunnel en virage par exemple), d'évaluer sa distance de visibilité pour permettre au conducteur, d'adapter la vitesse de circulation à fin de ne pas entraîner de «faux-positifs» systématiques et de diminuer l'impact global de l'incident.

A priori, trois phases d'exploitation de la ligne sont susceptibles de tirer bénéfice d'un dispositif de détection d'obstacles : l'ouverture de la ligne, la reprise en conduite manuelle après une panne et les mouvements en garage.

1.4.1 Ouverture de la ligne

Dans sa version actuelle, cette fonction est assurée par un train vide de passager qui ouvre l'exploitation de la ligne. Le train peut être conduit manuellement à des vitesses relativement basses et la sécurité du mouvement est assurée par le conducteur qui vérifie visuellement l'absence d'objet sur la voie. Sur le métro VAL de Lille, cette procédure n'a lieu qu'à titre exceptionnel (après des travaux sur la ligne, un évènement climatique particulier comme une tempête, ...). Sinon la reconnaissance de voie est assurée en automatique à vitesse réduite (train sans conducteur ni passager). Dans les deux cas, cette procédure longue et fastidieuse impacte d'autant plus l'exploitation que la ligne est longue.

La reconnaissance de la piste en automatique à basse vitesse reste un moyen de déterminer la présence des petits obstacles dans le plan de roulement de la voie, car la probabilité de trouver des obstacles dans un site propre intégral est très faible ainsi que la gravité des dommages en cas de collision compte tenu de la vitesse limitée du train. Pour que système optique apporte une amélioration, il devrait pouvoir détecter les petits obstacles avant de les percuter avec un taux de faux négatifs très faible, ce qui reste difficile en raison de la grande variété d'obstacles à détecter.

1.4.2 Reprise en conduite manuelle

Sur systèmes VAL, la conduite manuelle à vue est un mode dégradé du système de conduite automatique, quand le système de contrôle automatique du pilote automatique (PA) n'est plus actif. Il peut également être utilisé hors du domaine de contrôle automatique des trains comme la zone atelier et la voie d'essais. La reprise en manuelle de la rame s'opère dans les situations suivantes :

- Perte de contrôle de boucles sécuritaires de la rame (traverse chasse corps, portes...).
- Panne des équipements embarqués, d'automatismes.
- Panne des équipements fixes d'automatismes gérant un tronçon.
- Perte de localisation de la rame.

Ce mode permet de retirer une rame de l'exploitation lorsqu'un défaut interdit la poursuite de la conduite automatique. Il permet également aux rames de traverser des zones où le mode en automatique n'est temporairement plus possible. Dans ces cas, toute la sécurité repose sur l'agent de conduite présent à bord du train et sur l'application de procédures d'exploitation (échanges avec le Poste de Contrôle de Commande (PCC),...). Pour cette raison la consigne de vitesse en conduite manuelle à vue est limitée à 18 km/h (5 m/s).

L'agent de conduite doit respecter les consignes fournies par la signalisation latérale de la voie (signaux de manœuvre, limite de vitesse...).

À partir du pupitre manuel de conduite situé en tête du véhicule, l'agent d'exploitation a à sa disposition les commandes essentielles permettant de déplacer le train en conduite manuelle (traction, freinage), ainsi qu'un bouton d'arrêt d'urgence.

En cas de panne des automatismes embarqués, un agent d'exploitation doit :
- Se rendre sur place et reprendre le rame en conduite manuelle.
- Si le train était en panne entre deux stations, le conduire jusqu'à la station suivante pour évacuer les passagers.
- Reprendre sa marche sans arrêt jusqu'au terminus.

Actuellement, avec la limitation de vitesse à $18km/heure$, cette procédure engendre déjà une très forte perturbation sur la ligne qui voit sa vitesse commerciale chuter de $35km/h$ à moins de $18km/h$, surtout si l'incident a eu lieu peu après le départ d'un terminus. Or, pour respecter la norme EN 15227 en ne procédant à aucune modification ni sur la structure des caisses ni sur le système, la vitesse maximale de conduite à vue devrait être abaissée à $8km/h$, ce qui rendrait la ligne inexploitable jusqu'à ce que la rame soit évacuée.

1.4.2.1 Actions du système

Le système proposé dans nos travaux de thèse permet de reconnaître la voie à l'avant du train jusqu'à une distance D réévaluée en permanence. Au-delà de cette distance, le système fait l'hypothèse qu'une rame occupe la voie en aval de cette limite. Il vérifie alors que la vitesse courante du train est inférieure à la vitesse qui conduirait, en cas de déclenchement du freinage d'urgence, à

une collision à $8km/h$. Dès que cette condition n'est plus respectée, le système déclenche le freinage d'urgence.

Ce premier point permet de respecter strictement la norme, mais une collision, même à $8km/h$, n'est évidemment pas souhaitable ni même un déclenchement du freinage d'urgence. Cependant, afin de ne pas ralentir inutilement le train en présence d'un «faux positif» (la voie est libre mais non reconnue au-delà de la distance D), nous proposons de compléter la détection de la voie par une reconnaissance positive d'obstacles. Si un obstacle est identifié à la distance D, le système vérifie alors que la vitesse courant est inférieure à celle qui permet d'arrêter normalement le train en amont de l'obstacle.

La figure 1.4 nous montre un exemple de courbe de vitesse pour un cas où la voie n'est plus détectée et que le système de détection de trains ne reconnait pas la forme d'un train.

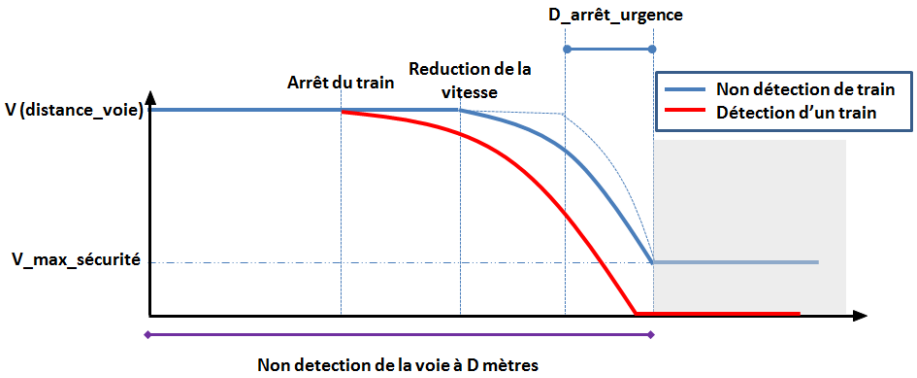


FIGURE 1.4 – Évolution de la vitesse du train en fonction du système proposé.

Par contre quand le système de détection de trains reconnait que la voie n'est plus visible, car elle est occupée par un train, la vitesse applique une décélération complète du train comme montré par le tracé rouge dans la figure 1.4. La recherche d'obstacle reposera sur l'utilisation d'un classifieur binaire pour reconnaître la silhouette frontale d'un train.

1.4.3 Mouvement en garage ou atelier

Actuellement, les garages sont des zones très chargées en équipements fixes d'automatismes. Il existe un grand nombre des balises pour assurer la localisation et signalisation du train pour pouvoir déplacer les trains en automatique, ce qui entraîne un sur-coût d'installation et de maintenance. D'autres garages nécessitent la présence d'un agent à bord pour pouvoir déplacer les trains en marche à vue, car il n'existe pas de contrôle automatique. Dans les deux cas, la vitesse des voitures est réduite et limitée à 8 km/h. Les niveaux de sécurité exigés dans ces déplacements ne sont pas comparables à la reprise en mode manuel, car il n'y a pas de passager à bord. Cependant, les dégâts du matériel ne sont pas souhaitables.

1.4.3.1 Actions du système

Pour ce type de conduite, et parce que dans les garages nous pouvons rencontrer l'hypothèse de fin de la ligne, les actions du système sur la vitesse du train sont les mêmes pour la non-détection de la voie que pour la détection d'un train. La figure 1.5 nous montre comme le train est arrêté à la distance D pour les deux cas.

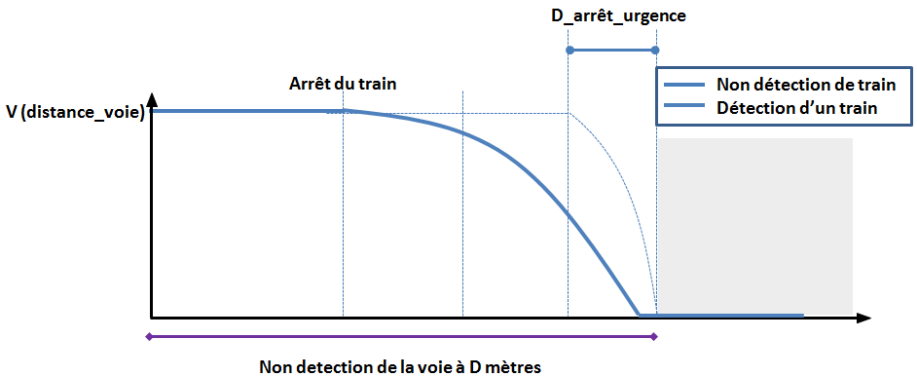


FIGURE 1.5 – Évolution de la vitesse en garage.

1.5 Objectifs

La thèse a pour ambition de mettre au service du métro automatique les avancées dans le domaine du traitement d'image. Les caméras sont aujourd'hui très peu présentes sur les matériels roulants ferroviaires, si ce n'est pour la vidéo surveillance. Dans ce cas, les images sont soit enregistrées à bord puis récupérées en fin de parcours ou en cas d'incident, soit transmises en temps réel au réseau du sol pour être visionnées à distance. Cependant, très peu de traitement intelligent de l'image est intégré dans les systèmes vidéo.

Pour répondre aux applications que nous visons, notre nouveau dispositif de détection d'obstacles se doit d'atteindre deux types d'objectifs :
- Objectifs techniques et performances à atteindre.
- Objectifs scientifiques.

1.5.1 Objectifs techniques et performances à atteindre

- En conditions normales de visibilité, le système doit détecter un train immobile/mobile situé devant lui suffisamment tôt pour que l'arrêt soit possible avant la collision.
- Le taux de «faux-négatifs» (non-détection d'un obstacle présent) doit être suffisamment faible pour permettre, combiné au ratio (durée de conduite en mode manuel/durée totale de circulation du train) et au «taux d'erreur humaine», d'atteindre en moyenne le niveau SIL4 sur l'ensemble des circulations du train. Ce taux est fixé à $10^{-5}/heure$ de fonctionnement en conduite manuelle.
- Le taux de «faux-positifs» (détection d'un obstacle à tort) doit être inférieur à $10^{-3}/heure$ applicable pour la conduite automatique en garages, la télé conduite et la conduite manuelle. Le dispositif devra être déconnecté pour la conduite automatique en ligne.
- L'évaluation de la distance à l'obstacle doit avoir une précision d'environ 10%.
- En visibilité réduite, qu'elle soit due aux conditions d'éclairage ou à la présence d'obstacles «normaux» (les parois du tunnel en virage par exemple), le système doit évaluer sa distance de visibilité pour permettre au conducteur dans le cas d'une reprise en manuelle, d'adapter la vitesse de circulation afin de ne pas entraîner de «faux-positifs» systématiques.

1.5.2 Objectifs scientifiques

- L'innovation proposée par ce projet repose sur la définition d'algorithmes prenant en compte la complexité de la zone filmée et surtout les taux très bas de faux positifs et négatifs. Actuellement, les résultats des systèmes de détection d'obstacles/reconnaissance des formes dans

le domaine du traitement d'images n'ont pas été traités de façon à atteindre des résultats satisfaisant l'exigence des niveaux de sécurité et disponibilité ferroviaire.
- Les caméras sont embarquées dans un véhicule ferroviaire roulant et le fond d'image est complexe. La vision avant du véhicule doit s'affranchir des différentes particularités d'un tunnel ferroviaire (courbes serrées, croisement d'un véhicule, tunnel monovoie, système d'aiguillage,...).
- La voie ferrée n'est pas une surface lisse (contrairement à la route) et l'environnement peut-être complexe et très variable comme en tunnel ou en aérien. Par contre, le fait que le véhicule soit guidé sera mis à profit pour améliorer notablement la fiabilité de la détection par la connaissance *a priori* de l'environnement.
- Les systèmes de détection d'obstacles/reconnaissance des formes dans le domaine du traitement d'images n'ont pas actuellement atteint des résultats 99.9999%.

Les systèmes de métros automatiques étant toujours en site propre intégral, les autres rames constituent presque exclusivement les obstacles potentiels à détecter. Pour atteindre l'objectif de sécurité dans tous les cas, la détection d'obstacles sera essentiellement basée sur la reconnaissance de la voie puisque la présence d'une autre rame sur la portion de voie surveillée, en interdira la reconnaissance.

Grâce à cette détection de la voie, nous pourrons définir une zone libre d'obstacles et une région d'intérêt pour déclencher une recherche d'obstacles. Une fois l'obstacle détecté, il faudra évaluer sa distance avec une précision compatible avec la vitesse du train.

Étant donné que la distance du train à l'objet détecté est une donnée critique, qui doit être supérieure à la distance d'arrêt du véhicule, nous proposerons des algorithmes permettant de réaliser cette fonction avec les niveaux de sécurité requis pour le système de transport ferroviaire.

Nous définissons comme «*bulle de sécurité*», la zone de la voie à détecter qui doit à tout prix être zone libre d'obstacles. Cette bulle est définie comme l'espace que le train va très probablement occuper dans les prochains instants, et dont il faut assurer la non-occupation par des obstacles. Un exemple de cette bulle de sécurité pour un train est exposé dans la figure 1.6.

De plus, cette approche peut être aussi utilisée dans les passages à niveau car si une voiture bloque ce dernier, les rails seront masqués par la voiture, et partant la zone de sécurité sera réduite. Cependant, comme dans notre cas le métro automatique est un TCSP intégral, nous ne traiterons pas les passages à niveau.

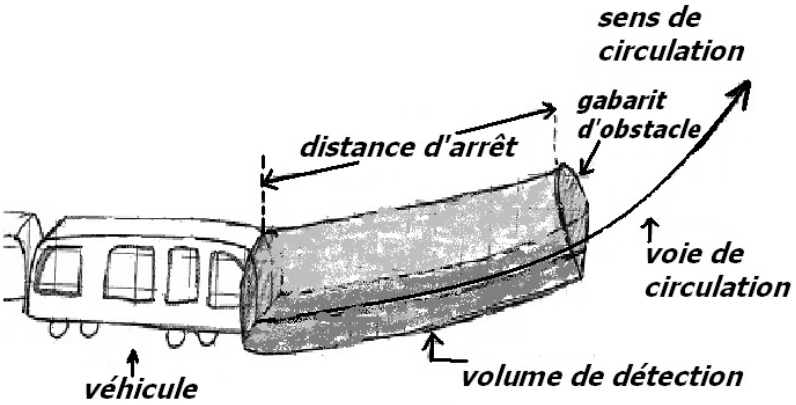


FIGURE 1.6 – Bulle de sécurité.

Cette étude permettra non seulement de déboucher sur des applications concrètes, mais surtout d'évaluer la maturité des technologies de traitements d'images pour réaliser des fonctions sûres appliquées aux systèmes ferroviaires.

1.6 Organisation du manuscrit

Dans le chapitre 2 nous exposons un état de l'art des systèmes existants, les principales caractéristiques de notre plateforme ainsi que le prototypage du capteur. Après son installation l'étalonnage de la caméra est analysé.

Le chapitre 3 est dédié à un état de l'art sur la détection de la voie ferrée, analysé à partir des exemples de différentes techniques testées. Aussi, nous présentons une approche originale pour la détection des rails avec l'objectif de déterminer une zone de la voie libre d'obstacles.

Le chapitre 4 présente la détection et la caractérisation des croisements ferroviaires, point important dans la détermination de la zone libre d'obstacles, car à l'arrivée d'un croisement divergent, le système doit être capable d'analyser les deux voies. Pour cela des techniques de détection et de classification sont proposées et évaluées à partir d'une base de données d'images. Déterminer la présence d'un train pour les appareils de voie convergents s'avère plus compliqué.

Nous présentons ensuite dans le chapitre 5 la détection des obstacles (trains). Comme indiqué, la détection de la voie nous permettra de déterminer une région d'intérêt dans laquelle nous pourrons réaliser une recherche des trains. Dans ce chapitre, nous exposons les démarches suivies pour obtenir un descripteur robuste et à la fois simple en calcul.

Finalement, le chapitre 6 est consacré aux conclusions et aux travaux futurs.

Chapitre 2

La plateforme mobile

Sommaire

2.1 État de l'art des systèmes de détection d'obstacles ferroviaires **15**
2.2 Choix du capteur **19**
2.2.1 Capteurs actifs 19
2.2.2 Capteurs passifs 20
2.3 Le Véhicule Automatique Léger : VAL **21**
2.3.1 Composition d'un véhicule 21
2.3.2 Caractéristiques 22
2.3.3 Modèle du véhicule 23
2.4 Étalonnage **24**
2.4.1 Modéliser une caméra 24
2.4.1.1 Vecteur de translation et matrice de rotation 25
2.4.2 Méthodes d'étalonnage 27
2.4.2.1 Détermination des paramètres intrinsèques 28
2.4.2.2 Obtention des paramètres extrinsèques 29
2.4.3 Méthode d'étalonnage proposée 32
2.4.3.1 Estimation du point de fuite 32
2.4.3.2 Filtrage temporel des points de fuite 35
2.4.4 Résultats 35

2.1 État de l'art des systèmes de détection d'obstacles ferroviaires

Dans cette section du chapitre, nous voulons exposer une description générale des principaux systèmes développés pour le milieu ferroviaire basés dans l'extraction de la voie et la détection d'obstacles. L'objectif est de comprendre les limitations des différentes méthodes et de tirer profit des systèmes déjà développés.

Détecteur d'obstacles multi-capteurs pour les voies ferrées

Dans le projet de [Mockel *et al.* 2003] implémenté et testé en 2003 en Allemagne, nous est présenté un système de détection d'obstacles sur la voie ferrée pour des trains régionaux. Le système est principalement basé sur la fusion de capteurs actifs et passifs, une approche qui conduit à une forte probabilité en termes d'efficacité, et permet de détecter des obstacles à 300 mètres avec une vitesse de 100 km/h dans des conditions normales (pas de pluie, pas de brouillard, etc).

Le système est composé de trois lasers et trois caméras, pour lesquels un capteur de chaque type a été réservé pour l'analyse proche et deux pour l'analyse à longue portée. Les caméras utilisées ont un grand champ de vision latérale et leur angle d'orientation est contrôlé par un miroir monté sur

un moteur pas à pas, grâce auquel les capteurs peuvent envelopper de manière efficace la voie dans les virages.

Le système utilise l'information externe concertant la position et la vitesse du train. En plus de cela, il dispose d'une base de données contenant des informations sur la forme de la voie et les objets qui la constituent.

Pour déterminer la position et forme de la voie, le système s'appuie sur l'extraction de voie à partir du traitement d'images et à partir des données de localisation du train et de la base de données de la voie ferrée.

Quant à la détection d'obstacles, le système utilise le traitement d'images et l'analyse des données lasers. Cette information provenant des capteurs est exploitée grâce à un filtre de Kalman.

Vingt-six obstacles en bois peint ont été placés sur la voie et sur les côtés avec des tailles entre $0.4\ m^2$ et $2\ m^2$. Dans cette configuration le système a démontré une grande résistance aux vibrations ainsi qu'un taux de faux négatifs très faible. Malheureusement, le projet ne nous fournit les valeurs de ces taux de détection.

Système de détection d'obstacles pour les trains automatisés

Le projet de Ruder, [Ruder *et al.* 2003] utilise la même plateforme que le projet précédent [Mockel *et al.* 2003] et parvient à détecter des obstacles de $0,4\ m^2$ jusqu'à une distance de 200 m. Cette fois-ci, les capteurs sont trois caméras et un laser. Une caméra avec une lentille de focale 35 mm est utilisée pour la détection lointaine et deux caméras en configuration stéréoscopique avec une focale de 12 mm pour une analyse jusqu'à 50 mètres. Comme capteur actif, le système possède un laser avec une portée de 150 mètres et un angle d'ouverture de 8 degrés. Comme information supplémentaire, il reçoit la vitesse et la position du train provenant du système de contrôle du véhicule et d'un GPS différentiel.

La détection d'obstacles par le traitement d'images est basée sur la fusion de différentes techniques :

- Détection des bords (*Edge elements*) : Après avoir appliqué un filtre de Canny [Parker 2010] à l'image certains agrégats de points peuvent être vus comme des obstacles.
- Discontinuité des rails (*Track gaps*) : Dans l'image de Canny, dès qu'il y a un morceau de rail qu'on ne voit pas, il est considéré comme un potentiel obstacle sur le rail.
- Flux optique (*Optical flow*) : Méthode utilisée pour la détection d'obstacles en mouvement.
- Analyse de la variance et de la corrélation en niveau de gris (*Grey value variance and correlation*) : la variance et la corrélation tout au long de la voie sont considérées comme invariantes. Un objet sur la voie changera ces valeurs. Pour avoir un meilleur point de vue de la voie, l'article propose d'utiliser la transformation vol d'oiseau (*Bird Eyes' View*).
- Analyse de la texture (*Statistics of textures*) : Cette approche est similaire à la précédente, mais appliquée à des portions de l'image plus petites. La variance de la voie est comparé avec celui de la variance des zones considérées comme obstacles possibles. Si le *cluster* obtenu est suffisamment large, cela est considéré comme un obstacle.
- Stéréovision et transformée de perspective *Stereo by inverse perspective mapping* ou *Bird Eyes' View* [Gangnet *et al.* 1984]. Deux caméras sont étalonnées avec le même système de coordonnées et partant la transformation des deux images doit être la même si tous les objets appartiennent au plan de roulement.

Pour la fusion de données, ils utilisent un filtre de Kalman : le filtre estime la localisation, la vélocité et l'accélération de chaque obstacle potentiel. La détection d'obstacles finale est réalisée par un système de décision et dépend des résultats obtenus par les différents systèmes proposés. Il existe une base de données dans laquelle sont enregistrés les différents objets qui appartiennent à la voie pour éviter les fausses détections.

Détection d'obstacles pour la voie ferrée

Développé au Japon, ce système [Yamashita *et al.* 1996] dispose comme hardware d'une caméra infrarouge et d'un laser. Cette caméra a été choisie pour avoir une bonne résistance aux conditions climatiques, car la caméra travaille dans la bande 8 à 13 μm, soit une des fenêtres de transmission de l'infrarouge, transparente au brouillard. Le système est divisé en trois sous-systèmes :

-L'estimation des voies ferrées : ce sous-système fait une prédiction de la voie ferrée. Pour y arriver, il reçoit différents types d'informations comme la position du train et les caractéristiques de la voie d'une base de données à partir desquelles il réalise une image artificielle de la voie pour faire une comparaison avec la voie détectée par le système de vision. L'extraction des rails est effectuée à partir de la caméra infrarouge donc, à cause des changements des caractéristiques des rails entre les images nuit/jour, deux extractions de la voie sont nécessaires : les rails apparaissent sombres en vision diurne et clairs en vision nocturne.

Pour déterminer la présence des rails, l'étude part de la considération qu'ils sont toujours continus dans la direction verticale de l'image, donc il est nécessaire d'extraire la moyenne horizontale avec un filtre de moyenne glissante. Avec un simple seuillage, les rails sont détectés.

-La détection d'obstacles en croisements se limite à la détection de trains. Celle-ci est assurée grâce à la grande différence en intensité entre le train et le fond de l'image avec une caméra infrarouge. La détection se réalise en extrayant des moyennes mobiles le long de la verticale et de l'horizontale de l'image. On obtient des portions d'obstacles dont le centre et la position sont calculés et envoyés au système de contrôle du laser pour la vérification de la distance.

-Le prototype dispose d'un système pour connaître les mouvements du train et permettre de faire le recalage entre l'image de la voie calculée et celle obtenue à partir de la caméra. Le laser permet de vérifier en même temps la distance des obstacles détectés par la caméra. Ce système actif est composé d'un laser et d'un gyromètre pour mesurer les mouvements du train.

Détection d'obstacles devant les véhicules de transport guidés par stéréoscopie

Ce système [Dooze *et al.* 1998, Louahdi *et al.* 2002] a été développé pour le métro automatisé VAL en 2001. Son principal objectif est de détecter tout objet placé devant la plateforme, qui posé sur la voie, dépasse le plan constitué par les pistes de roulement. Pour y arriver, le système propose d'utiliser des capteurs passifs.

Le capteur stéréoscopique est composé de trois caméras analogiques de 752 lignes par 582 colonnes. L'entre-axe de 0.4 m a été choisi pour minimiser l'erreur de localisation à une distance de 70 m. Les valeurs de la focale et de l'angle d'inclinaison des caméras sont différentes pour la caméra centrale et le stéréoscope.

Deux approches sont présentées : à partir du mouvement relatif de l'obstacle par rapport au train (*Ego motion*) et à partir de la mise en correspondance stéréoscopique. Comme la mise en correspondance est une opération très complexe qui nécessite un nombre élevé d'opérations arithmétiques, il est proposé d'utiliser seulement une ligne par image matricielle pour la détection d'obstacles. Cette ligne correspond à une droite de surveillance déterminée à la distance minimale d'arrêt. Le système a besoin d'un étalonnage pour s'assurer que la ligne choisie de chaque caméra corresponde bien au même point.

La procédure de détection d'obstacles dans l'entrevoie est basée sur un traitement des gradients des deux lignes des images gauche et droite, permettant l'identification d'un nombre de points d'intérêt correspondant aux bords des objets grâce à un seuillage. Un réseau de neurones de type Hopfield permet ensuite d'apparier les points d'intérêt retenus [Haykin 1994].

Le système réalise aussi la localisation et le suivi des pistes en utilisant la méthode des contours actifs [Kass *et al.* 1988] puisque les caractéristiques des pistes sont connues. En conséquence, le

modèle reste simple et compatible avec l'aspect temps réel requis.

Le procédé a été validé avec des expérimentations dans lesquelles figuraient différents types d'obstacles entre les voies. Le résultat des tests effectués avec différents objets, est une détection d'obstacles à une distance d'environ $30m$ avec une vitesse de 25 km/h.

Véhicule équipé d'un système de détection d'obstacles

Ce projet [Sasaki *et al.* 1999] dispose comme hardware de trois caméras CCD et d'un système de lampes pour éclairer la voie. La configuration des caméras permet d'avoir des distances de vue maximales de 100, 200 et 400 mètres. Le système change de caméra selon les conditions de visibilité, de la vitesse du train et de la forme de la voie.

De plus, le système reçoit des informations comme la vitesse et la localisation du train pour pouvoir synchroniser les bases de données de la voie et de l'équipement.

La détection d'obstacles est basée sur la reconnaissance des motifs à partir des images (*Pattern matching*) [Weiner 1973]. Pour éviter de confondre les équipements de voie avec des obstacles, il existe une base de données avec des échantillons (*templates*) de l'équipement. Avant de décider s'il s'agit d'un obstacle ou non, trois aspects sont à vérifier : sa localisation, sa position par rapport aux rails et sa signature (caractéristiques extraites de l'image). La figure ?? tirée de [Sasaki *et al.* 1999], montre un schéma de tous les systèmes qui coopèrent.

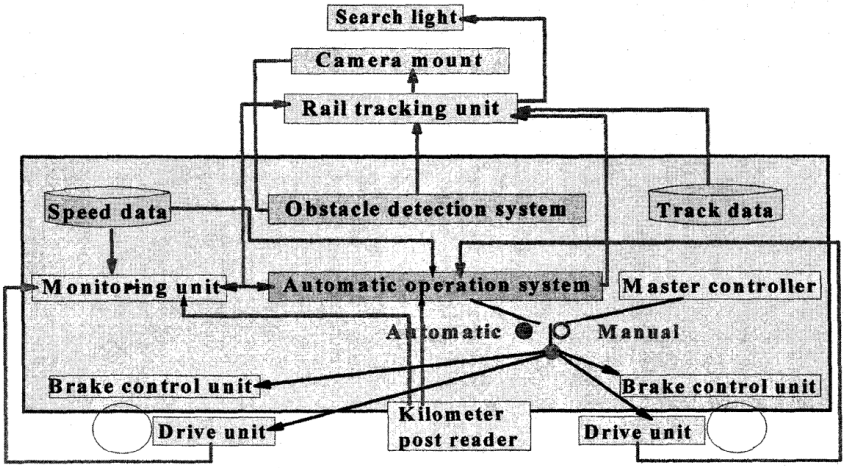


FIGURE 2.1 – Combinaison des systèmes pour aboutir au système de détection d'obstacles.

Détermination d'une zone libre d'obstacles pour les trains de maintenance

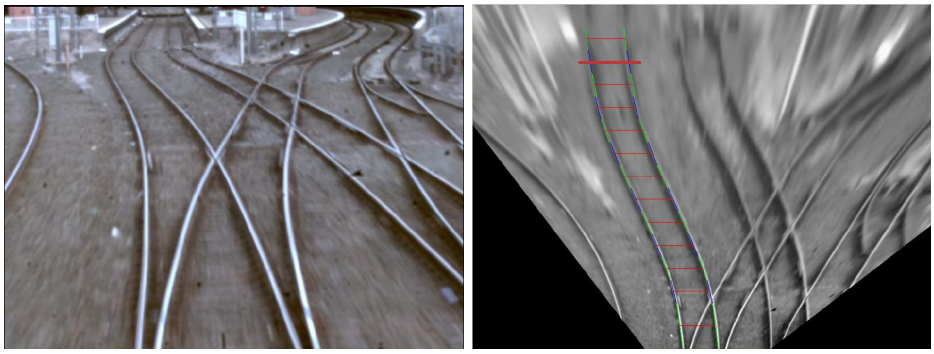
Dans cette étude de 2010 [Maire & Bigdeli 2010], Maire nous propose une approche basée sur la géométrie des voies ferrées,mais à partir de l'utilisation des images obtenues grâce à la transformée vue à vol d'oiseau (en anglais *Bird Eyes' View*) comme montré dans la figure 2.2.

Cette transformation est possible grâce à l'étalonnage de la caméra et la l'utilisation de l'hypothèse du monde plat.

La détection des rails est faite avec la transformée de Hough [Ballard 1981]. Pour limiter la charge de calcul, l'image originale est préalablement limitée aux contours des objets puis binarisée. Le système développé n'est pas temps-réel, et l'analyse des images est faite *a posteriori*. Comme le système utilise des séquences d'extérieur, la luminosité peut varier et par conséquent, pour déterminer le seuillage Canny, il est nécessaire d'indiquer la présence du rail dans l'image pour adapter

la valeur du seuillage. Le choix des rails est fait avec l'aide d'un opérateur et le système choisit les transformées de Hough les plus proches de la position indiquée par l'opérateur.

Par rapport à la détection d'obstacles, Le système détermine que la voie est libre tant que les gradients de Canny entre-voies sont perpendiculaire aux rails, c'est-à-dire, la détection des traverses.



(a) Image d'origine

(b) Image transformée et extraction des rails réalisée marquée en rouge et bleu

FIGURE 2.2 – Transformée en vue de vol d'oiseau et détection des rails

2.2 Choix du capteur

En ce qui concerne la localisation et la détection d'obstacles sur la voie ferrée, nous avons pu voir dans la section 2.1 qu'il existe plusieurs approches avec différents types de capteurs. Nous pouvons les classifier en deux grands groupes : les capteurs passifs tels que les systèmes de vision et les capteurs actifs tels que les radars et les télémètres lasers.

2.2.1 Capteurs actifs

Les capteurs actifs incorporent un émetteur, donc ils émettent de l'énergie pour éclairer la scène dans la bande spectrale du récepteur. Les plus utilisés sont les radars, les télémètres lasers et, beaucoup plus rarement, les ultrasons qui restent réservés aux obstacles proches, pour un contexte parking par exemple.

Nous n'étudierons pas les capteurs actifs en détail ici, dans la mesure où ils ne seront pas utilisés dans le système développé, par contre nous analysons les principales caractéristiques de trois capteurs actifs, leurs capacités et défaillances pour le système que nous cherchons à développer.

- Radar : Les radars sont utilisés en particulier dans les systèmes d'anti-collision et les systèmes de contrôle de vitesse adaptatifs (*Adaptative Cruise Control*, ACC) commercialisés depuis 1999 pour les voitures. L'avantage d'un radar réside dans sa grande portée (plusieurs dizaines voire centaines de mètres) et son indépendance face aux conditions météorologiques comme la pluie ou la neige [Hancock *et al.* 1998]. Sa précision peut être établie en fonction des caractéristiques de l'antenne.
  Cependant, le radar présente un certain nombre de limitations. Tout d'abord, il est très sensible aux pentes et aux virages et il a des difficultés à détecter les petits obstacles. Et surtout, l'utilisation du radar pose le problème des interférences électromagnétiques entre plusieurs véhicules lors de croisements dans la voie.
  D'autre part, l'étude de données radar laisse apparaître de nombreux échos parasites dans la chaussée ou sur les bords de route. Pour éviter que cela ne provoque un grand nombre de

fausses détections, les objets immobiles sont généralement filtrés en comparant la mesure de leur vitesse à la vitesse du véhicule équipé [Perrollaz 2008].

- Télémètre Laser : Les lasers fonctionnent sur le même principe que les radars, mais utilisent un faisceau lumineux. Ils sont très utilisés dans les domaines de la robotique pour la navigation et cartographie des robots en terrain inconnu [Steux & El Hamzaoui 2010]. La télémétrie laser arrive à détecter les obstacles et à connaître leurs distances [Espino *et al.* 2011]. La détection est possible sur tout type d'objets, métalliques ou non, mais les objets sombres posent des problèmes, de même que les surfaces transparentes qui sont traversées par le faisceau laser sans réflexion.
  Le laser est le meilleur candidat pour la détection d'obstacles, car il a une très bonne résolution et une bonne portée, sauf en présence de brouillard (en fonction de la longueur onde), de pluie ou de neige, car les faisceaux sont dispersés et/ou absorbés.
  La poussière qui se dépose sur les lentilles frontales est source de problèmes surtout dans un environnement ferroviaire (ballast, tunnels, etc.) et pose des contraintes de maintenance. De plus, le laser a une faible résistance aux chocs.
- Sonar : Le sonar est très limité en portée (environ *2 m*), en vitesse et en résistance à l'environnement. Il est sensible à la température et à la pression et également aux autres systèmes utilisant la même fréquence (lors des croisements entre trains). Par conséquent, son utilisation est restreinte, se limitant principalement à l'aide au stationnement. Dans le domaine ferroviaire il faut ajouter que les métros génèrent des ultrasons en provenance des moteurs et des freins, ce qui rend difficile l'usage des ultrasons comme possibles capteurs à exploiter.

2.2.2 Capteurs passifs

Un capteur passif est un capteur qui reçoit de l'énergie émise sans que lui-même n'irradie la scène. Ces capteurs, dans le monde de la robotique, sont majoritairement des caméras. Deux approches sont alors possibles : l'utilisation d'une seule caméra ou l'exploitation de plusieurs points de vue avec plusieurs caméras reliées de manière rigide sur le véhicule.

Dans notre cas, après avoir fait une analyse des avantages et inconvénients des principaux capteurs actifs, des besoins et contraintes d'exploitation ferroviaire et des coûts, nous avons choisi d'utiliser une seule caméra couleur dans le spectre visible. En outre, nous avons choisi une caméra couler car, il est envisageable que l'utilisation des caméras continue à se répandre de plus en plus sur les trains pour d'autres applications ferroviaires comme la télé-conduite, la détection des feux de signalisation, la vidéo surveillance des passages à niveau, etc.

Le choix d'un capteur de vision est motivé par plusieurs raisons. Tout d'abord, il permet une perception globale de la scène, étant le seul capteur capable de fournir des informations à la fois sur la voie ferrée courante, comme les panneaux et les feux de signalisation, les autres voies de circulation et les obstacles. De plus, la détection d'obstacle par traitement d'images est privilégiée, car son déploiement sur des trains n'augmentera pas de façon significative le coût global du système. Enfin, il s'agit d'un système peu encombrant, il a un caractère non intrusif et il consomme relativement peu d'énergie.

Le système final doit être assez robuste pour s'adapter aux différents états et changements d'environnement, d'éclairage, ainsi qu'à des conditions météorologiques variables. De plus, il doit être fonctionnel en tunnel ou à l'extérieur et également être capable d'accepter des changements rapides dans la luminosité.

La dynamique est l'une des principales caractéristiques d'un capteur d'images. Elle définit en quelque sorte la plage de luminosité que le capteur peut mesurer. La valeur exprimée en dB, est basée sur le rapport entre la plus forte et la plus faible luminosité. Une caméra vidéo standard n'est capable de capturer qu'une partie de l'information contenue dans une scène de gamme dynamique élevée. Cependant, un capteur avec une grande dynamique peut acquérir des images contenant des

zones sombres de la même image sans être ébloui par la forte luminosité. Un capteur de 60 dB, par exemple, peut acquérir l'image d'une scène dans laquelle la luminosité maximale est 1000 fois plus forte que la valeur plus faible ($20 \times \log 1000 = 60\ dB$)

La stéréovision a été écartée, en raison de son coût et car la reconstruction 3D de la scène ne nous a pas semblé indispensable pour détecter les obstacles. De plus, son étalonnage et son entretien induisent un sur-coût supplémentaire. L'approche monovision consiste en une interprétation de l'image avec une connaissance *a priori* de ce qu'est un obstacle (généralement un véhicule ou piéton) et non une reconstruction *3D* de l'environnement.

2.3 Le Véhicule Automatique Léger : VAL

Cette brève description permettra de recenser les principaux systèmes et l'équipement des trains VAL et comprendre les besoins et contraintes du système à développer.

La caractéristique principale du VAL est d'atteindre, en site propre, une vitesse commerciale (vitesse moyenne, utile à l'usager tenant en compte la vitesse de pointe et les arrêts en stations) de 35 km/h à une fréquence élevée (jusqu'à 1 minute). Son gabarit qui autorise une insertion aisée en milieu urbain, avec des viaducs étroits, des tranchées et des tunnels réduits. Le roulement sur pneu amenuise bruit et vibrations et permet de gravir des pentes jusqu'à 12 % [Verhille 2007].

2.3.1 Composition d'un véhicule

La longueur totale du VAL est de $26,14\ m$ pour une largeur de $2,08\ m$ et une hauteur de $3,275\ m$. Il est composé de deux voitures non scindables supportées chacune par deux essieux moteurs. Les figures 2.3 et 2.4 montrent les différentes cotes du train.

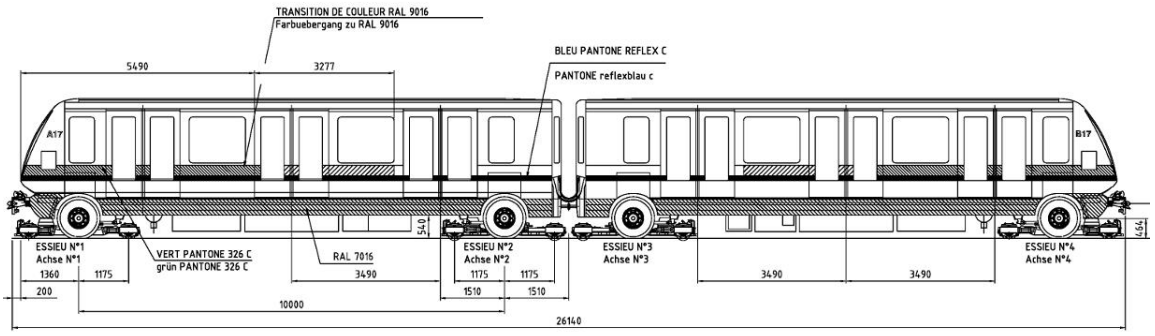


FIGURE 2.3 – Plan détaillé de la vue latérale du VAL 208

Le roulement du VAL est composé de deux essieux par voiture, séparés de 10 m, dont les roues porteuses sont à enveloppe pneumatique et roulent sur une piste de roulement spécifique dont la largeur est adaptée à celle des pneumatiques.

Chaque essieu se compose de :
- deux roues porteuses,
- un pont réducteur différentiel,
- deux disques de frein,
- quatre roues horizontales de guidage.

Le guidage en ligne est assuré par les roues de guidage montées aux quatre extrémités du cadre de guidage. Pour cela, tout au long de la piste, en plus des deux pistes de roulement, il existe deux

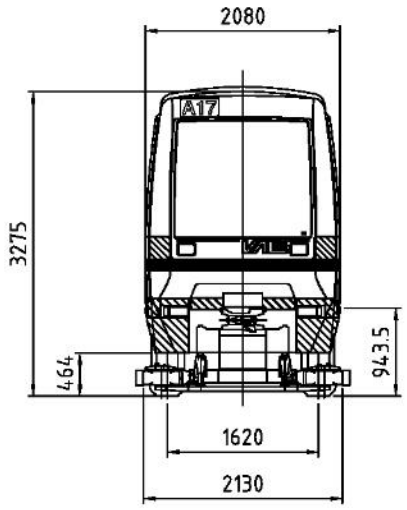


FIGURE 2.4 – Plan détaillé de la vue frontale du VAL 208

rails de guidage. En l'absence des rails de guidage, ce qui arrive lors des changements de direction sur les appareils de voie (aiguillages), le guidage est assuré par deux galets situés l'un à l'avant et l'autre à l'arrière de chaque essieu. Ces galets s'insèrent dans un rail central de guidage composé d'une ornière dans l'axe de la voie. Une explication plus détaillée des aiguillages sera exposée dans le chapitre 4.

2.3.2 Caractéristiques

Le tracé de la voie est constitué d'une succession d'alignements droits et d'arc de cercle de rayon minimum 150 mètres en ligne (sauf cas particulier) et de 40 mètres en zone de garage atelier.

Le tracé est défini avec les conditions de confort suivantes :

- Le rayon de courbure horizontal minimal sur la ligne (courbure de la voie) est de 40 m.
- Le rayon vertical minimal sur la ligne est de 600 m et la pente maximale de 7%.
- Le dévers de la voie va de 0% à 13% avec une variation du dévers maximale de $0,55\%/m$.
- Variation normale du dévers < 4 %/s
- Il existe différents types d'aiguillages (convergents et divergents qui peuvent être associés en "cœur-croisé") dont il faut tenir compte.
- La voie peut être unique ou double.

Après avoir analysé les spécificités ferroviaires et le métro léger VAL, ses principales caractéristiques de fonctionnement sont les suivantes :

- La plage de décélération en freinage d'urgence garantie est de 1.8 à 2.4 m/s^2.
- La plage de vitesse du véhicule va de 0 à 80 km/h.
- Les vitesses moyennes en exploitation sont de 25 km/h dans les aéroports et de 36 km/h dans les villes.
- Pour une reconnaissance de la voie (avant ouverture de la ligne le matin) le tour de la ligne est effectué à 10 m/s (36 km/h).
- La vitesse de fonctionnement en mode dégradé en conduite manuelle est limitée à 18 km/h. Pour les nouvelles installations cette vitesse est limitée à 8 km/h.
- Le roulis de la caisse oscille entre $+/-3°$, mais en fonctionnement normal la vitesse angulaire du roulis est limitée à $< 2°/s$.

- Le tangage de la caisse oscille entre $+/-5°$, mais en fonctionnement normal la vitesse angulaire du tangage est limitée à $< 2°/s$.
- Le lacet de la caisse oscille entre $+/-0,2°$.

2.3.3 Modèle du véhicule

La modélisation dynamique d'un véhicule est une étape importante pour la conception du système de contrôle. La dynamique d'un véhicule est un système complexe et dépendant des nombreux paramètres. De façon simplifiée, le modèle dépend des forces créés entre les roues et le sol et les différentes pièces mécaniques du véhicule.

En tenant compte du contexte de notre étude, à savoir la détection d'obstacles par vision ce rapport ne cherche pas à modéliser de façon détaillée la dynamique du véhicule, mais à obtenir une relation entre les positions réelles des obstacles et leurs positions dans l'image. Pour cela, les mouvements relatifs de la caisse (lacet, roulis, tangage) qui porte la caméra par rapport à l'essieu doivent être pris en compte.

Puisque les vibrations auxquelles est soumis le train sont faibles, nous pouvons faire l'hypothèse que l'axe de l'essieu et celui de la caméra sont perpendiculaires. Une représentation de ces axes est montrée dans la figure 2.5 [Ross 2012].

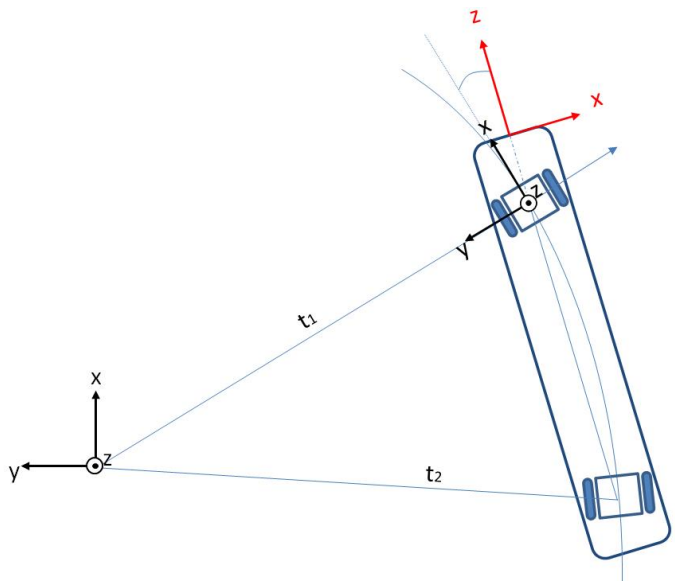


FIGURE 2.5 – Modèle du véhicule val et axes des essieux et de la caméra

À partir du modèle présenté, il est possible de démontrer que l'axe d'orientation de la caisse du train est défini par le vecteur $\vec{t_1} - \vec{t_2}$. Les vecteurs $\vec{t_1}, \vec{t_2}$ représentent la position de l'essieu avant et de l'essieu arrière par rapport au repère global de la scène. Dans les virages, il existe un angle (θ_{caisse}) entre l'axe de la caisse et celui de la voie au droit du pivot d'essieu. θ_{caisse} est défini par l'équation 2.1. À chaque instant, il faut tenir compte de cet angle pour pouvoir relier les images de la caméra à sa projection dans le monde réel.

$$\theta_{caisse} = \theta_T - \angle(\vec{t_1} - \vec{t_2}) \tag{2.1}$$

où θ_T représente l'angle de $\vec{t_1}$.

2.4 Étalonnage

L'étalonnage de la caméra est une des premières étapes de la vision par ordinateur surtout lorsque des mesures de distances précises sont nécessaires. Des exemples d'applications sont : la reconstruction dense des points à partir de la stéréovision [Kolmogorov & Zabih 2002], l'inspection visuelle [Mar *et al.* 2011], la localisation d'objets [Lefaudeux & Nashashibi 2012] ou la localisation de la caméra dans l'espace [Liu *et al.* 1990].

Cet étalonnage peut être divisé en deux phases. La première est la modélisation de la caméra en approximant mathématiquement le modèle physique et optique, et la deuxième consiste à estimer les valeurs de ce modèle.

2.4.1 Modéliser une caméra

La modélisation d'une caméra est basée sur la définition d'un ensemble de paramètres qui se rapproche le plus possible du comportement physique du capteur à l'aide d'un modèle géométrique. Nous proposons d'utiliser le «modèle sténopé», du nom du simple dispositif optique (en anglais : *pinhole*), car il permet de modéliser un appareil photographique et c'est le plus couramment utilisé [Dominique & Eric 1998].

Le modèle géométrique sténopé (figure 2.6) est constitué d'un plan, appelé «plan image», sur lequel l'image se forme suite à une projection perspective. Un point O_c, placé à une distance fixe f ou focale du plan image, est appelé le centre optique (ou point focal) et est tel que les rayons de lumière reflétés par les objets passent par ce point, formant une image perspective de la scène sur le plan image.

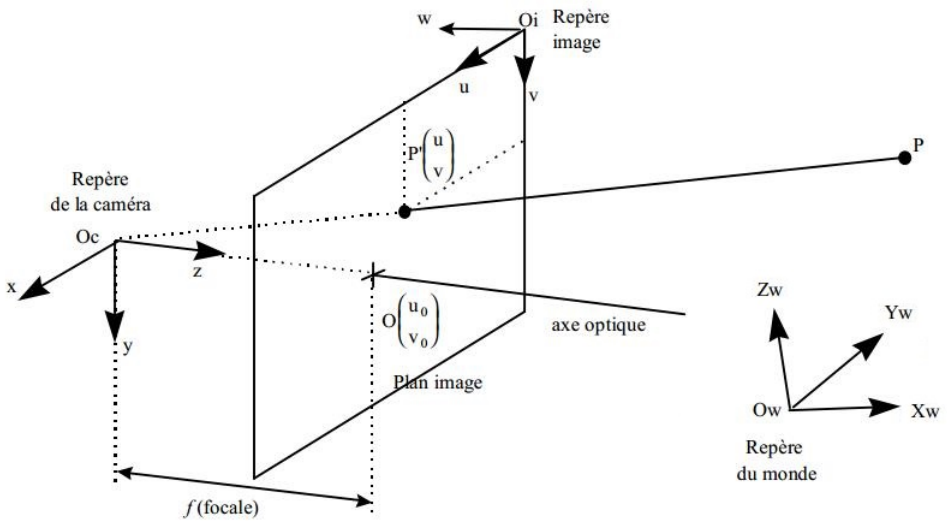


FIGURE 2.6 – Illustration du modèle sténopé.

Les caméras sont caractérisées par deux groupes de paramètres, d'un côté les paramètres intrinsèques, qui modélisent la géométrie interne et les caractéristiques optiques de la caméra (focale, facteurs d'agrandissement, distorsion de la lentille, etc.) et de l'autre les paramètres extrinsèques qui lient position et orientation du centre optique de la caméra (repère du capteur) au repère du système global localisé de la scène. Ces derniers permettent de localiser la caméra dans l'environnement et sont composés de trois paramètres pour modéliser la translation (T_x, T_y, T_z) entre O_c (repère de la

caméra) et O_w (repère de la scène) et trois pour la rotation (α, β, γ). Dans notre cas, le repère de la scène est placé dans le plan de roulement et centré sur l'axe de la voie et la caméra est placée derrière le pare-brise à l'avant du train comme présenté dans la figure 2.7.

On rappelle qu'étant donné le modèle sténopé d'une caméra, la relation entre les coordonnées homogènes du point P sur l'image et ses coordonnées homogènes dans le repère de la scène (O_w) s'exprimera par l'équation 2.2. La première matrice A exprime l'intégralité des paramètres intrinsèques du capteur caméra, et la seconde B exprime l'intégralité des paramètres extrinsèques.

$$\begin{pmatrix} s_u \\ s_v \\ s \end{pmatrix} = \underbrace{\begin{pmatrix} \alpha_u & 0 & u_0 & 0 \\ 0 & \alpha_v & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}}_{A} \underbrace{\begin{pmatrix} R & T \\ 0 & 1 \end{pmatrix}}_{B} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \tag{2.2}$$

Avec :
- $(X, Y, Z, 1)$, les coordonnées homogènes du point P dans le repère O_w.
- R, la matrice de rotation entre le repère de la scène et la caméra.
- T, le vecteur de translation entre la référence de la scène et la référence de la caméra.
- A, la matrice de projection de perspective. Elle exprime la transformation entre le repère de la caméra et celui de l'image.
- (u_0, v_0), projeté du centre optique O_c sur le plan image, exprimé en pixels.
- α_u, α_v définis comme suit : $\alpha_u = f \cdot k_u$ et $\alpha_v = f \cdot k_v$ où k_u et k_v sont le facteur d'échelle horizontal et vertical $(pixels/mm)$.
- f, la distance focale.
- (s_u, s_v, s) les coordonnées homogènes de la projection du point P sur l'image respectivement.

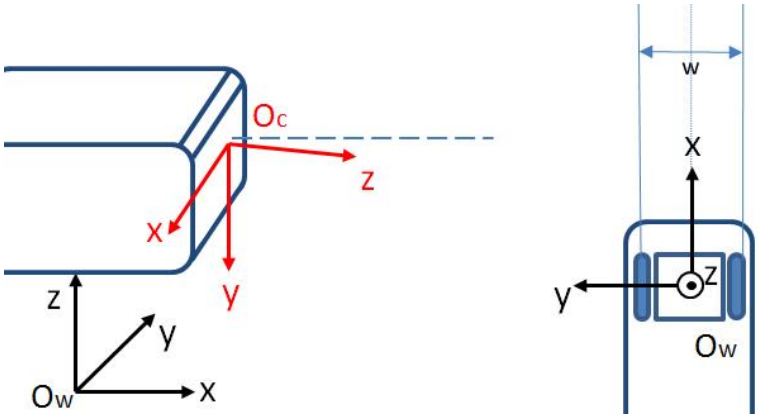


FIGURE 2.7 – Axes des coordonnées de la scène et de la caméra.

2.4.1.1 Vecteur de translation et matrice de rotation

Afin de prendre en compte les déplacements que la caméra va effectuer, nous devons choisir un repère statique dans la scène. Nous introduisons donc un repère scène O_w, (figure 2.7). Les positions des points $3D$ et de la caméra sont alors décrites par rapport à ce repère. Nous pouvons établir une relation entre les points $3D$ du O_w et le repère O_c, connaissant le vecteur de translation de la caméra $T = [T_x,\ T_y,\ Tz]'$, et l'orientation de la caméra exprimée par une matrice de rotation de taille 3×3.

Notre système optique sera placé à l'avant du train, derrière le pare-brise à une hauteur et une inclinaison fixes. À partir de l'analyse des caractéristiques exposées dans la section 2.3.2, les

changements d'angles dus aux vibrations du train seront considérés comme nuls.

Pour définir cette matrice de rotation, nous utiliserons la convention «X, Y, Z» correspondant aux rotations de la caméra autour de chaque axe pour arriver à la position des axes. Cette séquence est définie comme suit :

$$R(\gamma_1) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\gamma_1) & \cos(\pi/2+\gamma_1) \\ 0 & \cos(\pi/2-\gamma_1) & \cos(\gamma_1) \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\gamma_1) & -\sin(\gamma_1) \\ 0 & \sin(\gamma_1) & \cos(\gamma_1) \end{pmatrix} \tag{2.3}$$

$$R(\gamma_2) = \begin{pmatrix} \cos(\gamma_2) & 0 & \cos(\pi/2-\gamma_2) \\ 0 & 1 & 0 \\ \cos(\pi/2+\gamma_2) & 0 & \cos(\gamma_2) \end{pmatrix} = \begin{pmatrix} \cos(\gamma_2) & 0 & \sin(\gamma_2) \\ 0 & 1 & 0 \\ -\sin(\gamma_2) & 0 & \cos(\gamma_2) \end{pmatrix} \tag{2.4}$$

$$R(\gamma_3) = \begin{pmatrix} \cos(\gamma_3) & \cos(\pi/2+\gamma_3) & 0 \\ \cos(\pi/2-\gamma_3) & \cos(\gamma_3) & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} \cos(\gamma_3) & -\sin(\gamma_3) & 0 \\ \sin(\gamma_3) & \cos(\gamma_3) & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{2.5}$$

$$R = R_1(\gamma_1)R_2(\gamma_2)R_3(\gamma_3) \tag{2.6}$$

où :

- (R_1) représente la rotation d'un angle γ_1 autour de X
- (R_2) représente la rotation d'un angle γ_2 autour de Y
- (R_3) représente la rotation d'un angle γ_3 autour de Z

Si le tangage est défini par (θ), le roulis par (ψ) et le lacet par (ϕ), la série de rotations qui en résulte s'écrit : $R_N^B = R_a R_1(\theta) R_2(\phi) R_3(0)$

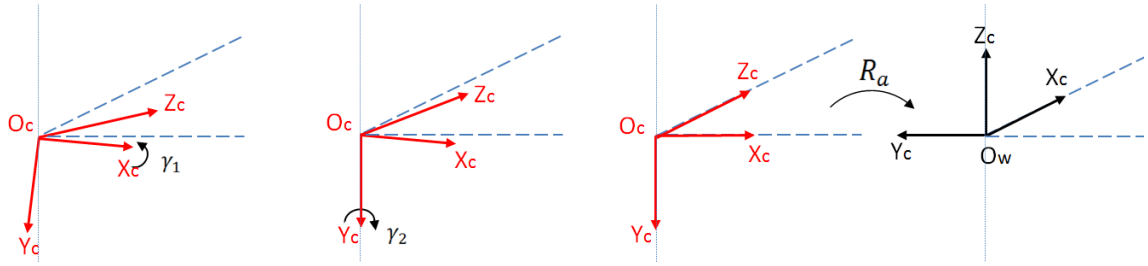


FIGURE 2.8 – Rotations suivies pour adapter le repère de la caméra R_C au repère de la scène R_W

$$R_1(\theta) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) \\ 0 & \sin(\theta) & \cos(\theta) \end{pmatrix} \tag{2.7}$$

$$R(\phi) = \begin{pmatrix} \cos(\phi) & 0 & \sin(\phi) \\ 0 & 1 & 0 \\ -\sin(\phi) & 0 & \cos(\phi) \end{pmatrix} \tag{2.8}$$

$$R_3(0) = I \tag{2.9}$$

$$R_a = \begin{pmatrix} 0 & 0 & 1 \\ -1 & 0 & 0 \\ 0 & -1 & 0 \end{pmatrix} \tag{2.10}$$

Le roulis sera considéré nul, car notre système sera implanté dans une cabine de métro et nous avons considéré que l'effet du roulis de la caisse est faible et donc l'influence sur la transformation du pixel en distances est négligeable.

Cette hypothèse peut être envisagée seulement si le roulis du train est faible. Dans le cas d'un roulis excessif, comme le train est considérablement plus haut qu'une voiture, on a un mouvement latéral de la caméra qui ne permet pas de définir le système de coordonnées.

Un exemple de l'effet du roulis dans la projection de la voie devant la caméra est montré dans la figure 2.9.

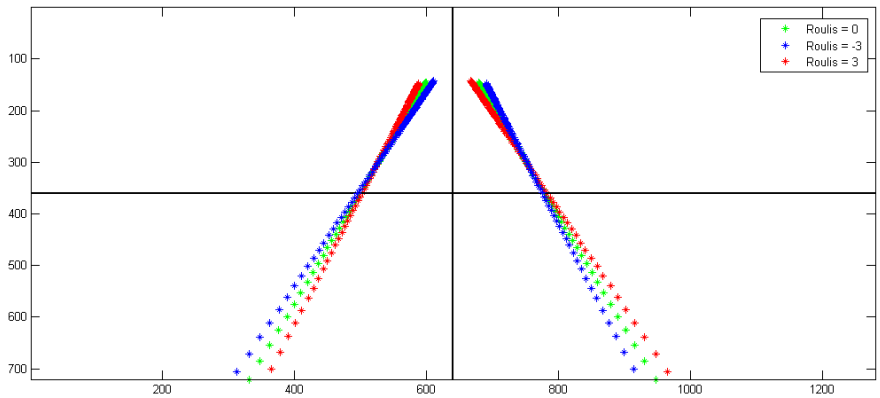


FIGURE 2.9 – Effet du roulis

La séquence des rotations est alors : $R_N^B = R_a R_1(\theta) R_2(\phi)$.

La relation entre les points dans le repère de la caméra R_C et le repère de la scène R_W est définie par 2.11 et 2.12.

$$\begin{bmatrix} P_w \\ 1 \end{bmatrix} = \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} P_c \\ 1 \end{bmatrix} \tag{2.11}$$

$$\begin{bmatrix} P_c \\ 1 \end{bmatrix} = \begin{bmatrix} R^T & -R^T T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} P_w \\ 1 \end{bmatrix} \tag{2.12}$$

2.4.2 Méthodes d'étalonnage

Après la description du modèle de caméra, il est important d'accéder aux valeurs des paramètres du modèle. C'est le rôle de la phase d'étalonnage, étape nécessaire dans le domaine de la vision par ordinateur, ceci afin de pouvoir extraire l'information $3D$ à partir des images $2D$.

L'étalonnage de la caméra est un point important du système. La connaissance des repères 2D (plan image) et 3D (monde réel) et la relation existante entre eux sont nécessaires pour la détection d'obstacles. L'objectif est de pouvoir déterminer la distance de l'obstacle à partir de sa localisation dans l'image. Cette relation *a priori* simple dépend cependant de l'orientation de la caméra par rapport à la voie, donc de l'assiette de la voiture porteuse de la caméra et du gradient de pente sous la voiture et dans toute la zone d'intérêt.

Les méthodes d'étalonnage peuvent être classées selon différents critères : linéaires ou non-linéaires ; implicites ou explicites, auto-étalonnage, etc. Une très bonne analyse des avantages et inconvénients des différentes méthodes peut se trouver dans [Salvi *et al.* 2002]. L'objectif de cette

thèse n'est pas d'analyser en profondeur l'étalonnage des caméras monoculaires, nous exposerons ici deux approches : avec mire d'étalonnage ou sans mire.

L'étalonnage classique utilise une mire plane. Sachant que chaque point de la mire se projette dans l'image suivant le modèle décrit dans l'équation 2.2, les paramètres peuvent être obtenus grâce à une résolution par la méthode des moindres carrés. Cette technique utilise une mire plane dont les coordonnées bidimensionnelles sont connues et plusieurs prises de vue de la mire (déplacements inconnus de la caméra ou de la mire) pour établir une relation entre les homographies grâce aux propriétés de la géométrie projective et aussi obtenir les paramètres de la caméra. De plus, elle permet d'aboutir à un étalonnage très fin.

Il existe plusieurs outils développés pour obtenir les paramètres de la caméra comme le *«caméra Calibration Toolbox»* de Caltech [Bouguet 2006] développé pour *Matlab* et *OpenCV*.

Les techniques d'auto-étalonnage ne nécessitent l'utilisation d'aucune mire d'étalonnage. Elles se basent essentiellement sur la géométrie épi-polaire (matrice fondamentale et matrice essentielle). L'idée consiste à déplacer la caméra dans une scène statique et la rigidité de la scène fournit les caractéristiques nécessaires. Il faut déterminer des points caractéristiques dans chacune des scènes et, dans un premier lieu, nous calculons des correspondances de points entre deux images. Ceci peut être fait manuellement, mais il existe aussi des méthodes automatiques. Nous calculons ensuite l'homographie en utilisant les images elles-mêmes [Sturm 2008].

2.4.2.1 Détermination des paramètres intrinsèques

La caméra sera équipée avec un objectif à focale fixe que nous avons déterminée après avoir «dimensionné» l'ensemble du système *caméra + objectif*. Pour cela, nous avons pris en compte le champ de vision de la caméra et une résolution de l'image à une distance D pour pouvoir voir les rails. Comme la caméra existe, nous avons accès à ses spécifications pour déterminer aisément la focale souhaitée.

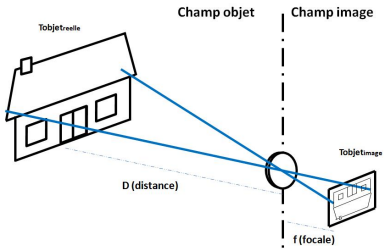


FIGURE 2.10 – Approximation de la distance objet/caméra supérieure à $3m$

Grâce à la figure 2.10, nous pouvons déduire facilement à partir du théorème de Thalès, la focale qui permet d'avoir, à pleine échelle, un objet de taille connue T_{objet} à la distance D. Ainsi, la focale est donnée par 2.13 :

$$f = \frac{T_{image} D}{T_{objet}} \tag{2.13}$$

Dans notre cas, la focale a été définie à 16 mm pour que la projection du rail à 25 mètres ait une résolution de 30 pixels.

Une fois la focale de l'objectif définie et les caractéristiques du capteur connues, en appliquant le modèle de caméra complet sans distorsion de l'équation 2.2, tous les paramètres intrinsèques sont connus.

2.4.2.2 Obtention des paramètres extrinsèques

Dans la configuration finale de notre système, les paramètres extrinsèques pourront être connus *a priori*, car l'installation de la caméra sera faite pour avoir un champ de vision défini. Sa position et son orientation seront fixées pour que les rails soient toujours présents dans l'image. Certes, ses paramètres peuvent être inexacts, car installer la caméra avec un tel degré d'orientation (θ) n'est pas toujours facile. De plus, la pression des pneumatiques varie pour les différents trains ce qui ferait varier les paramètres pour chaque train.

Pour obtenir les paramètres extrinsèques, les systèmes basés sur l'utilisation d'une mire impliquent la réalisation de manœuvres dans les ateliers avec un train immobile ce qui implique des tâches laborieuses pour la mise en fonctionnement et la maintenance du système. Dans notre cas, la mire doit avoir de très grandes dimensions. La figure 2.11 montre un test d'étalonnage avec une mire de dimensions $1\ m \times 0.7\ m$ placée devant le train à une distance d'environ 10 mètres. Ce test montre que cette taille n'est pas suffisante pour étalonner la caméra.

FIGURE 2.11 – Capture d'image avec une mire d'étalonnage devant le train sur le plan de roulement

En conclusion de ce test et étant donné des inconvénients futurs que la mire induit dans l'exploitation ferroviaire, toutes les méthodes utilisant une mire ont été écartées.

Une bonne approche serait de déterminer de façon automatique ces paramètres, à partir de l'environnement ferroviaire, ce qui permettrait d'avoir des valeurs exactes d'étalonnage pour chaque train de façon indépendante en fonction de son état (crevaison, caméra déplacée, ...). Grâce à la connaissance de la morphologie de la voie, il peut être envisagé d'effectuer l'étalonnage en temps réel à partir des images elles-mêmes.

Par conséquent, la méthode proposée pour déterminer ces paramètres est basée sur une estimation robuste du point de fuite de l'image grâce à la détection des rails dans une portion de voie droite [Nieto *et al.* 2007b]. Ce point de fuite est alors utilisé pour estimer la rotation et la translation de la caméra par rapport à la voie R_w. Grâce à cette méthode, la caméra peut être étalonnée de façon simple, sans ajouter de contraintes importantes à l'exploitation des trains. Il est même possible de réaliser *«on-line»* les vérifications sur la caméra.

À partir du point de fuite, il est possible de définir le tangage (θ) et le lacet (ϕ) de la caméra par les équations 2.14 et 2.15. α_v et α_h représentent le champ de vision, u_{vp} et v_{vp} la position du point de fuite dans le plan image et M et N le nombre de pixels horizontalement et verticalement.

$$\theta = \arctan\left[\tan\alpha_v\left(1 - \frac{2v_{vp}}{N}\right)\right] \tag{2.14}$$

$$\phi = \arctan\left[\tan\alpha_h\left(\frac{2u_{vp}}{M} - 1\right)\right] \tag{2.15}$$

En développant l'équation 2.2 et en considérant la projection des points P_w pour lesquels $X \to \infty$, le point de fuite peut être aussi obtenu par les équations 2.16 et 2.17 :

$$u_{vp} = fk_u \tan\phi + u_0 \tag{2.16}$$

$$v_{vp} = fk_v \tan\theta \sec\phi + v_0 \tag{2.17}$$

Les paramètres de translation, entendus comme la position de la caméra par rapport à l'origine de coordonnées globale (axe de la voie), sont définis par la distance entre l'essieu et l'avant de la caisse et la hauteur de la caméra par rapport au plan de roulement.

Dans le cas d'une version définitive, ces paramètres seront fixes et par conséquent la position de la caméra sera connue *a priori*, mais dans la configuration de notre prototype la position de la caméra par rapport au premier essieu est inconnue, car les enregistrements ont été faits avec une caméra installée avec un trépied à l'intérieur de la rame.

Cet inconvénient nous oblige à déterminer la position de la caméra à partir des enregistrements. Nous avons des inconnues dans nos équations que ne nous permettent pas d'obtenir les paramètres extrinsèques. Ces inconnues sont la hauteur de la caméra et son déplacement latéral par rapport à l'axe principal du train.

Pour obtenir la hauteur de la caméra et le déplacement latéral par rapport à l'essieu sans utiliser de mire, c'est-à-dire, à partir de la scène, nous proposons d'utiliser la mesure de l'écartement des rails dans l'image. Pour cela, nous utilisons les propriétés des deux lignes parallèles dans l'espace et leur projection dans l'image [Wang *et al.* 2007].

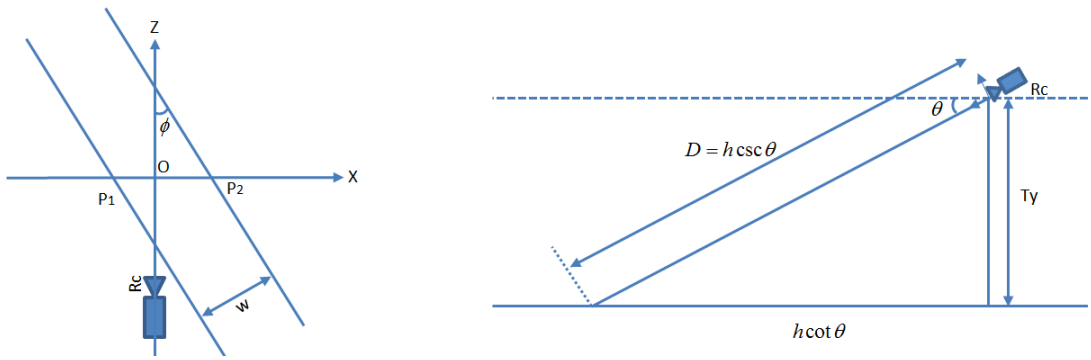


FIGURE 2.12 – Systèmes de coordonnées utilisé dans l'image et leurs caractéristiques dans le monde pour étalonner la caméra et déterminer sa hauteur à partir de l'écartement W des rails.

Si nous analysons la figure 2.12, nous pouvons déduire que la distance entre le centre optique de la caméra O_C et l'objet, dans notre cas(le plan de roulement) peut se déduire de l'équation 2.18

$$D = h \csc\theta \tag{2.18}$$

D'autre part, la taille de l'objet T_{objet} égale à la distance P1P2, est donné par l'équation 2.19 :

$$T_{objet} = w \sec\phi \tag{2.19}$$

Si nous appliquons ces deux équations (2.19 et 2.18) dans l'équation 2.13, nous pouvons obtenir la hauteur de la caméra à partir de l'écartement des rails et de la focale de la caméra comme montre l'équation 2.21 :

$$f = \frac{T_{image} h \csc\theta}{w \sec\phi} \tag{2.20}$$

$$h = \frac{f w \sec\phi}{T_{image} \csc\theta} \tag{2.21}$$

Pour obtenir la position latérale de la caméra, nous pouvons considérer deux cas : la caméra est placée exactement au centre de la voie $T_y = 0$, ou bien la caméra est décalée d'une distance T_y comme le montre la figure 2.13.

FIGURE 2.13 – Deux hypothèses du placement de la caméra. a) La caméra est placée au centre de la voie. b) La caméra est décalée du centre de la voie d'une distance T_y

Quand il n'existe pas de translation entre la caméra et le centre de la voie, en développant 2.12, nous pouvons déduire que la projection du centre de la voie ($Y = 0$, quelque soit X) ne dépend que du tangage de la caméra et correspond à l'équation 2.22. De plus, cette valeur coïncide avec le point de fuite des rails comme montre l'équation 2.16.

$$u - u_0 = -k_u f \frac{Xc}{Zc}\bigg|_{y=0} = -k_u f \frac{-X\cos\theta\sin\phi + T_z\sin\phi\sin\theta}{X\cos\theta\cos\phi - T_z\sin\theta\cos\phi} = k_u f \tan\phi \tag{2.22}$$

où u représente la position horizontale en pixels, u_o projeté du centre optique O_c sur le plan image, exprimé en pixels, ku le facteur d'échelle horizontal en $(pixels/mm)$, (θ) le tangage et (ϕ) le lacet.

Cependant, quand la caméra est déplacée par rapport au centre de la voie, pour ($Y = 0$, $\forall X$) le développement de l'équation 2.22 nous montre comme présenté dans l'équation 2.23, que la projection du centre de la voie se voit déplacé par l'effet de la translation T_y de façon proportionnelle à la projection de cette dernière.

$$u - u_0 = -k_u f \frac{Xc}{Zc}\bigg|_{y=0} = -k_u f \frac{-X\cos\theta\sin\phi + T_z\sin\phi\sin\theta + T_y\cos\phi}{X\cos\theta\cos\phi - T_z\sin\theta\cos\phi + T_y\sin\phi} = k_u f(\tan\phi + \frac{T_y\cos\phi}{Zc}) \tag{2.23}$$

En conséquence, il est possible d'obtenir T_y à partir de la position du centre de la voie u_c dans l'image et la position où elle devrait être si la caméra était au centre de la voie u_{vp}. Cette position est obtenue à partir du point de fuite de notre image, car même si la caméra est décalée, la position du point de fuite reste la même, (équation 2.24).

$$u - u_0 = \lim_{X\to\infty} -k_u f \frac{Xc}{Zc}\bigg|_{y=0} = -k_u f \frac{-X\cos\theta\sin\phi + T_z\sin\phi\sin\theta + T_y\cos\phi}{X\cos\theta\cos\phi - T_z\sin\theta\cos\phi + T_y\sin\phi} = k_u f\tan\phi \quad (2.24)$$

$$u_{vp} - u_c = k_u f\tan\phi - k_u f(\tan\phi + \frac{T_y\cos\phi}{Zc}) = -k_u f\frac{T_y\cos\phi}{Zc} \quad (2.25)$$

2.4.3 Méthode d'étalonnage proposée

Au vu des équations détaillées précédemment, les paramètres extrinsèques de notre système peuvent être déterminés *a posteriori* en exploitant les caractéristiques physiques de la voie. Pour cela, nous avons défini une méthode d'étalonnage que suit le diagramme de la figure 2.14. Nous proposons de calculer le point de fuite pour une séquence de M images où le métro avance en ligne droite. Cela nous permettra d'affiner le résultat en faisant un moyennage tout au long de la séquence pour les différents points de fuites obtenus. Pour la détection du point de fuite, nous proposons d'utiliser la Transformée de Hough [Ballard 1981], donc il est nécessaire de réaliser une extraction de contours de l'image avec un filtre de Canny [Canny 1986]. Puis, une fois obtenu le point de fuite, nous pouvons déterminer l'écartement des rails au centre de l'image (v_o). Ceci sera réalisé pour un nombre M d'images consécutives avec le train en mouvement, pour finalement, réaliser obtenir la moyenne des valeurs obtenus. Une étape importante est de détecter des résultats erronés pour obtenir une valeur de point de fuite et d'écartement des rails corrects.

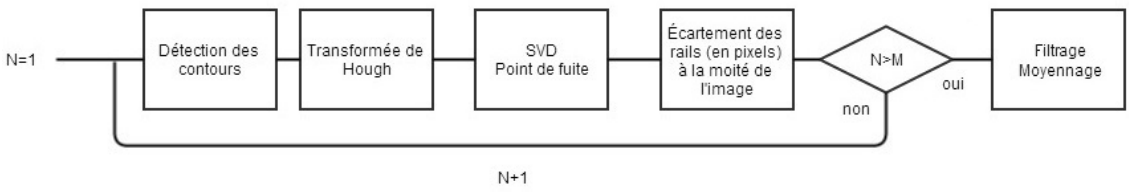


FIGURE 2.14 – Étapes de la méthode proposée pour l'étalonnage de la caméra.

2.4.3.1 Estimation du point de fuite

Il existe des méthodes permettant d'extraire des informations 3D à partir d'une seule image. Il s'agit d'indices monoculaires comme les ombres, les occlusions, etc. Cependant, le plus évident reste l'étude des «lignes de fuite» et des «points de fuite» (*vanishing lines* et *vanishing points* en anglais).

On peut définir le point de fuite dans une image comme étant le point de l'espace vers lequel toutes les lignes de profondeur, ou fuyantes se rencontrent, ou semblent converger.

En termes de caméra, le point de fuite V d'une droite l (de l'espace 3D) est l'intersection du plan image de la caméra avec un rayon parallèle à l passant par le centre de projection de la caméra. Une conséquence de ce qui précède est que des droites parallèles au plan image apparaissent parallèles sur l'image (figure 2.15).

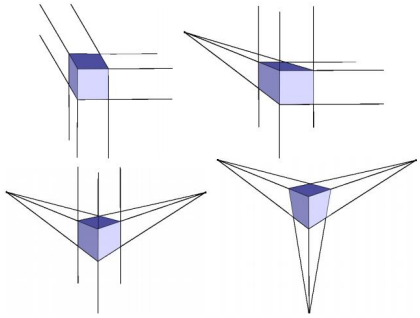


FIGURE 2.15 – Nombre de points de fuite variable en fonction du point de vue

En pratique, il existe plusieurs approches permettant de calculer des points de fuite. Le plus simple étant de calculer l'intersection de deux droites de l'image qu'on sait parallèles dans l'espace 3D.

Dans le cas de deux droites parallèles l_1 et l_2, leur point de fuite est $V = l_1 \times l_2$. Dans le cas où l'on dispose de n droites parallèles ($n > 2$), on sait que V appartient à toutes ces droites, ce qui se traduit par : $l_1^T V = l_2^T V = ... \; l_n^T V = 0$.

Ce système peut se résoudre au sens des moindres carrés en utilisant une décomposition en valeurs singulières (*SVD* de l'anglais *Singular Value Decomposition*). Il est préférable de toutes les utiliser pour augmenter la précision du calcul de V.

Pour détecter les lignes présentes dans l'image, nécessaires pour le système d'équations, on propose utiliser la très connue transformée de Hough, qui permet de détecter dans l'image la présence de courbes paramétriques appartenant à une famille connue (droites, cercles, ellipses) [Ballard 1981].

La transformée de Hough

Elle a été décrite pour la première fois par Paul Hough en 1962 et est utilisée comme technique de reconnaissance de formes appliquée sur des images numériques. Les applications de la transformée de Hough sont nombreuses et variées, telles que la détection des traits du visage sur des images, la reconnaissance des codes à barres ou d'empreintes digitales.

Le fonctionnement de la transformée de Hough se base sur l'utilisation d'un espace paramétrique (espace de Hough) permettant de simplifier le problème complexe de détection globale des formes dans l'espace image. Dans ce nouvel espace, la détection est locale et donc plus simple. De cette façon, une droite passant par un point (x, y) dans l'espace image peut être définie par son équation polaire (équation 2.26), avec le couple (θ, ρ), les paramètres de l'espace de Hough.

$$x * \sin\theta + y * \cos\theta = \rho \tag{2.26}$$

Une fois n lignes détectées, on peut définir le système d'équations 2.27 et le résoudre comme indiqué précédemment avec SVD pour obtenir la solution v du système.

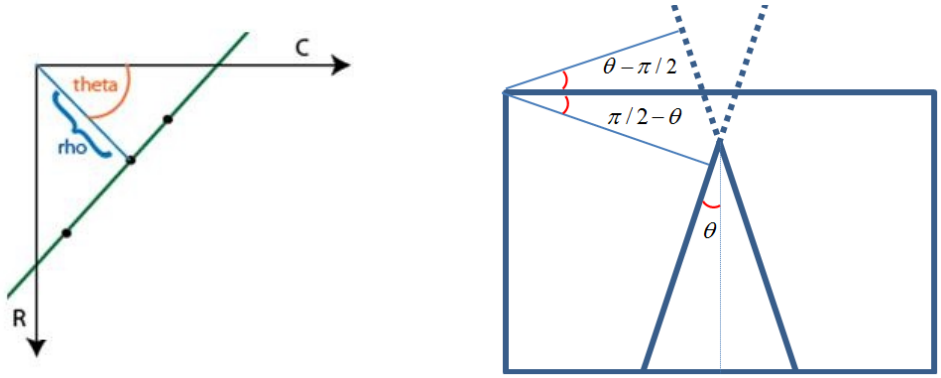
$$[c \mid s] * v = p \tag{2.27}$$

où $v = (x, y)^T$, $c = (\cos\theta_0, ... \cos\theta_{n-1}, \cos\theta_n)^T$, $s = (\sin\theta_0, ... \sin\theta_{n-1}, \sin\theta_n)^T$ et $p = (\rho_0, ...\rho_{n-1}, \rho_n)^T$.

Avant de réaliser la recherche des lignes dans l'espace de Hough, nous proposons de réaliser un filtrage dans l'espace des paramètres en fonction de l'orientation des droites dans l'image. Le métro

présente des environnements très structurés ce qui d'un coté offre des avantages pour définir des critères de recherche. En contrepartie, ces environnements comportent beaucoup de structures avec des segments droits, ce qui introduit des détections erronées.

L'objectif est de supprimer les lignes présentes dans l'image qui n'appartiennent pas au groupe des «lignes de fuite» du point v recherché. Ce filtrage est basé sur l'orientation θ des lignes appartenant au plan de roulement (rails, tapis, etc.) et consiste à supprimer toutes les lignes qui n'ont pas une orientation proche de celle recherchée (figure 2.16).

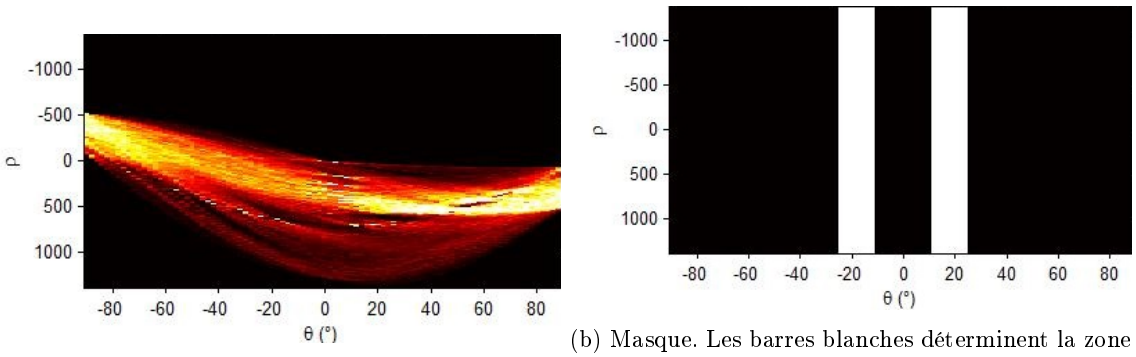


(a) (b)

FIGURE 2.16 – Paramètres ρ et θ en fonction de l'orientation des lignes. À droite on peut voir la projection du plan de roulement et les valeurs de θ que devront avoir ces lignes dans l'espace de Hough

L'objectif est de supprimer dans l'espace de Hough toutes les lignes de l'image que n'ont pas l'orientation des lignes du plan de roulement. Pour cela, la recherche de lignes dans l'espace de paramètres sera exclusivement réalisée dans deux zones définies en fonction de l'orientation du point de fuite de l'image. Ce point de fuite sera initialement introduit manuellement pour une vidéo en ligne droite. Ces zones de recherche sont définies de la manière suivante :

- $\theta < \pi/2 - \theta_0(1 + 10\%)$
- $\theta > \pi/2 - \theta_0(1 - 10\%)$
- $\theta > \theta_0(1 - 10\%) - \pi/2$
- $\theta < \theta_0(1 + 10\%) - \pi/2$



(a) Transformé de Hough pour une image de la voie.

(b) Masque. Les barres blanches déterminent la zone de l'espace de Hough dans laquelle on réalise la recherche des lignes

FIGURE 2.17 – Filtrage dans l'espace de Hough.

L'inconvénient de la transformée de Hough est que c'est un algorithme très lourd en temps de calcul, mais grâce au rétrécissement proposé de l'analyse dans l'espace de Hough ce défaut se voit réduit.

Une fois que les lignes ont été détectées dans l'espace de Hough, nous pouvons obtenir le point de fuite correspondant au groupe de lignes détectées grâce à la décomposition en valeurs singulières.

2.4.3.2 Filtrage temporel des points de fuite

Avant de réaliser la moyenne temporelle des points obtenus, on doit supprimer les valeurs aberrantes. Ces points de fuite peuvent être dus à l'introduction de lignes détectées dans l'espace de Hough qui n'appartiennent pas aux lignes de profondeur de notre scène, mais qui ont une orientation similaire.

D'autre part, comme notre plateforme est soumise à des vibrations, le fait d'utiliser une séquence vidéo pour détecter le point de fuite nous permettra d'obtenir un résultat plus fin car l'effet des vibrations sera moyenné sur les différentes images du train en mouvement.

Pour filtrer, nous pouvons utiliser différentes méthodes comme celle présentée dans les travaux de [Suttorp & Bucher 2006], qui propose d'utiliser un filtre passe-bas suivi d'un filtre de Kalman. Néanmoins, nous choisissons une méthode statistique qui nous permettra de pondérer chaque point de fuite détecté V_i. La méthode choisie est le «score standard» (*Z-score* en anglais) et représente la différence entre une détection X et la moyenne μ de la distribution des points, exprimée en unités d'écart type σ selon la formule 2.28.

$$Z = \frac{X - \mu}{\sigma} \tag{2.28}$$

Le *Z-score* donne l'écart relatif entre la valeur mesurée et la valeur cible.

La standardisation consiste à remplacer X par $Z = (X - \mu)/\sigma$ et donnera systématiquement une loi normale centrée réduite pourvu que X soit normale.

Une propriété importante de la distribution normale, exposée dans la figure 2.18, est l'intervalle de confiance dans cette distribution selon lequel :
- Environ 68% des valeurs se retrouvent dans $[-\sigma; \sigma]$
- Environ 95% des valeurs se retrouvent dans $[-2\sigma; 2\sigma]$
- Environ 99,72% des valeurs se retrouvent dans $[-3\sigma; 3\sigma]$

Les vibrations du train, essentiellement dues aux irrégularités de la voie, suivent une loi normale. Ces vibrations étant de faible amplitude, on peut considérer que la distribution des points de fuite suit également une loi normale (en 2D) bien que la relation qui les lient à la position de la caméra ne soit pas strictement linéaire. Nous avons défini un seuil pour supprimer les points de fuites aberrants en base aux pourcentages exposés précédemment. Ce seuil nous permettra de supprimer tous les points de fuite qui ont un *Z-score* supérieur ou égal à 1 et inférieur ou égal -1. Partant, la moyenne ne sera réalisé qu'avec les valeurs telles que le *Z-score* satisfait $|Z| < 1 \Rightarrow \mu - \sigma < X < \mu + \sigma$.

2.4.4 Résultats

Les différentes séquences vidéo que nous avons pu enregistrer à bord d'un métro VAL pour ce projet ont été réalisées avec deux caméras et en deux occasions différentes. La première fois 4 séquences ont été acquises un jour de pluie à l'aéroport d'Orly et la deuxième séquence un jour ensoleillé à Rennes avec une autre caméra. Les caractéristiques des caméras sont exposées dans le tableau 2.1. Les caméras utilisées lors de ces enregistrements n'étaient pas celle retenue pour le dispositif.

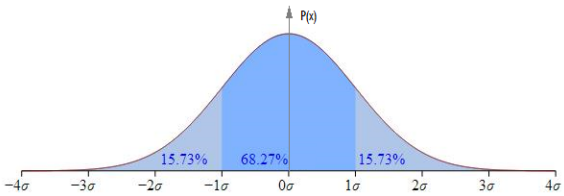


FIGURE 2.18 – Courbe de distribution normale et pourcentages des valeurs qui se retrouvent par section.

Tableau 2.1 Paramètres intrinsèques pour les différentes séquences

Paramètres	Vidéo 1 à 4	Séquences Rennes
Focal (mm)	6.2	7.1
s_u (μm)	5.76	8.8
s_v (μm)	4.29	6.6
N (pixels)	480	1080
M (pixels)	620	1920

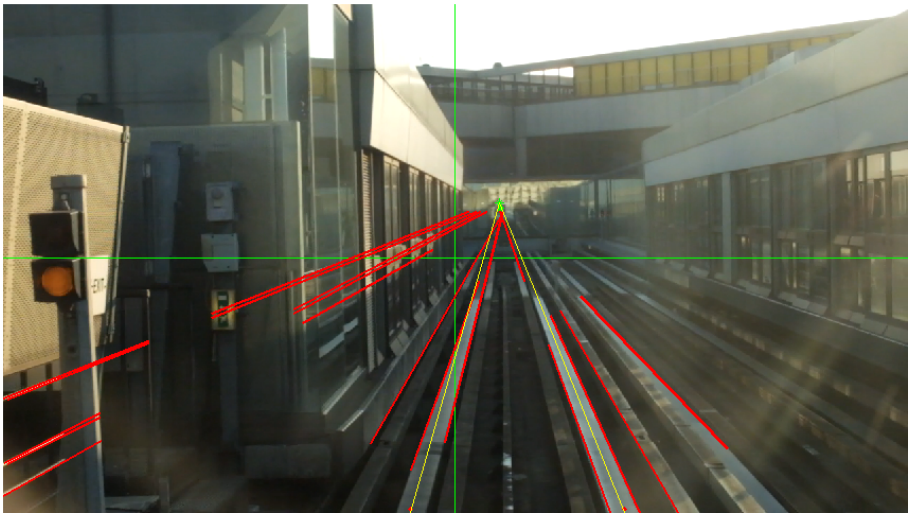
Pour les deux caméras, les résultats correspondant aux paramètres extrinsèques sont exposés dans le tableau 2.2. Les résultats sont cohérents puisque la paire de vidéos 1 et 2, et 3 et 4 a été réalisée dans deux trajets différents. Dans chaque trajet les séquences ont été enregistrées avec un trépied à l'intérieur du train, placé au milieu de la rame derrière le pare-brise. Cependant, à cause des reflets de la vitre à l'intérieur des tunnels, les enregistrements vidéo ont été arrêtés, ce qui a pu produire des faibles changements dans les angles d'orientation, et aussi à cause des vibrations de la caisse, le trépied a pu se déplacer.

Par rapport à la séquence de Rennes, où une autorisation d'enregistrement a été obtenue, la valeur acquise de la hauteur de la caméra correspond aux estimations faites surplace. La caméra a été place cette fois-ci sur une barre horizontale présente dans la rame derrière le pare-brise et sa position par rapport au plan de roulement est connue. Rappelons que la formule pour déterminer la hauteur est en fonction de l'orientation de la caméra.

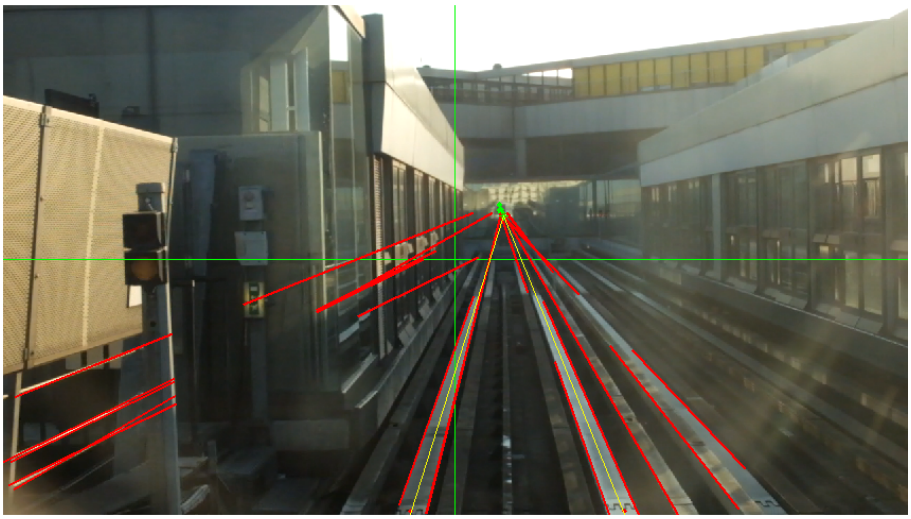
La figure 2.19 nous montre un exemple de l'extraction des différents points de fuite pour une séquence vidéo en ligne droite. La sous-figure (d) montre le filtrage temporel réalisé grâce au *Z-score*, ou les points en bleu représentent les points de fuite écartés pour le calcule de la moyenne.

Tableau 2.2 Résultats d'étalonnage obtenus pour les différentes séquences

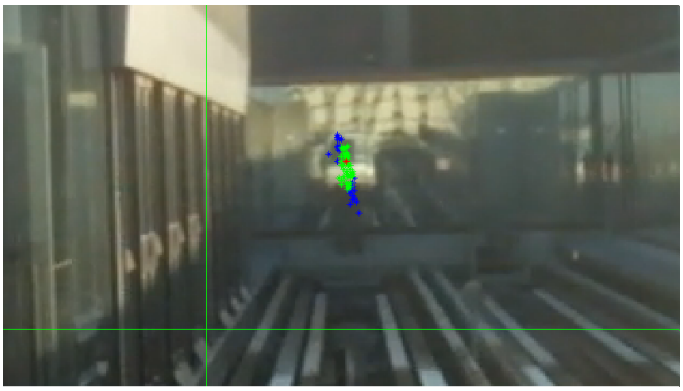
Paramètres	Vidéo 1	Vidéo 2	Vidéo 3	Vidéo 4	Séquences Rennes
Tangage (θ)	-10.932°	-11.33°	-9.493 °	-9.774°	-9.1917°
Lacet (ϕ)	2.2445°	2.825°	2.1614 °	2.078°	-1.5161°
Hauteur (h) (m)	2.2329	2.248	2.2605	2.2305	2.3701
Translation (T_y) (m)	0.01	-0.005	-0.025	-0.010	0.009



(a) Détection des lignes du point de fuite.



(b) Détections des lignes avec des lignes erronées (*outliers*).



(c) Ensemble des points de fuite détectés. En bleu les points rejetés par le filtre statistique. En rouge la moyenne des points verts.

FIGURE 2.19 – Exemple de la détection du point de fuite.

Chapitre 3

Détection de la voie

Sommaire

3.1 Motivations **39**
3.1.1 Base de données 39
3.1.2 La piste de roulement 40
3.2 État de l'art et approches testées **41**
3.3 Approche retenue **51**
3.3.1 Sélection du Masque 51
3.3.2 Analyse des gradients 53
3.3.3 Recherche du modèle dans l'espace 3D 57
3.3.3.1 Nombre d'itérations de RANSAC 58
3.3.3.2 Score du modèle 59
3.3.3.3 RANSAC pour chaque rail 61
3.3.4 Dépendance temporelle 62
3.3.5 Détermination de la distance maximale de détection 63
3.3.5.1 Effets des changement de pente 64
3.4 Résultats **67**

3.1 Motivations

Les trains n'ont pas besoin de réaliser l'extraction et le suivi des rails pour contrôler leur trajectoire puisqu'il s'agit d'un système guidé. Par contre, il existe deux raisons pour lesquelles la détection des rails est intéressante :

- Avoir une région d'intérêt pour l'exploitation des capteurs : Nous cherchons à développer un système temps réel dont la minimisation de la zone de recherche permettra d'obtenir des résultats plus rapides et robustes.
- La seconde raison est le fait que les obstacles ont une connexion avec le sol, il y aura donc une discontinuité dans les caractéristiques du sol.

Dès lors, le fait que le véhicule soit guidé sera mis à profit pour améliorer notablement la fiabilité de la détection par la connaissance *a priori* de l'environnement. Si nous arrivons à reconnaître les rails jusqu'à une distance D à l'avant du train, alors la voie est libre d'obstacles sur cette zone et il y a potentiellement un obstacle au-delà. Pour cela, nous avons donc évalué les techniques présentées dans les sections 3.2 et 3.3 afin d'analyser la faisabilité d'un système de détection des rails dans le contexte de sécurité ferroviaire.

3.1.1 Base de données

Bien que pour la détection des lignes ou des panneaux routiers il existe des bases de données publiques comme *Kitti Road Dataset* [Fritsch *et al.* 2013] ou *German Trafic Sign Recognition*

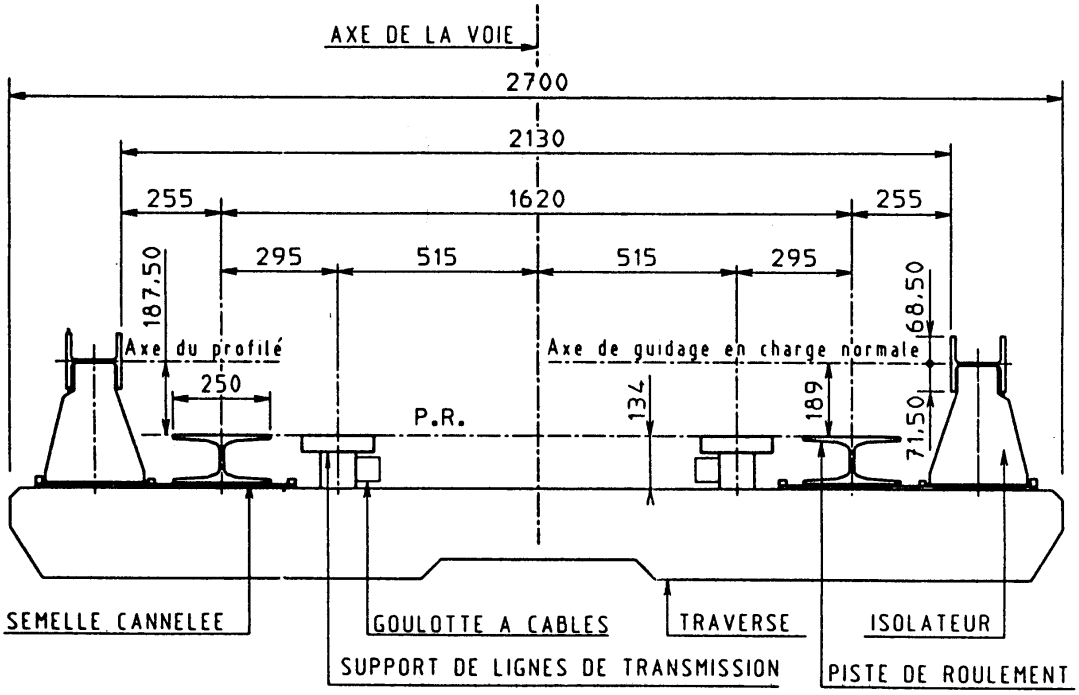


FIGURE 3.1 – Coupe transversale de la voie du VAL

Benchmark [Stallkamp *et al.* 2011] actuellement, aucune base de données publique n'existe pour la détection des rails, ce qui permettrait de comparer notre approche à la littérature. La base de données utilisée dans ce projet a été acquise par nos propres moyens dans une portion du trajet de la ligne du métro automatique OrlyVal, qui relie Antony et l'aéroport OrlySud, avec un parcours de $7,8$ *km* dans différentes conditions de voie : aérien, tunnel et en viaduc.

3.1.2 La piste de roulement

La voie du VAL est constituée principalement des éléments suivants :
- Deux pistes de roulement.
- Deux barres de guidage latéral, servant également à l'alimentation électrique du matériel roulant, montées sur des isolateurs verticaux, des blochets ou des traverses en béton armé.
- Des appareils de voie à guidage central (aiguillages).
- Des passerelles et des trottoirs d'évacuation.
- Des chemins de câbles (tapis à câbles en voies au sol) appelés aussi tapis.

L'ouvrage au sol est constitué d'un radier béton sur lequel sont disposées les voies, le cheminement et tous les éléments nécessaires au fonctionnement de la ligne Orlyval. Il est bordé de part et d'autre, d'une clôture (grillage) afin d'empêcher toute intrusion sur le site et assurer la définition de «site propre» (section 1.2.1).

Un aperçu de la coupe transversale de la voie avec ses mesures est montré dans la figure 3.1.

Les pistes de roulement servent de support au véhicule, elles sont de type métallique d'une largeur de 25 *cm* et avec une hauteur de 12 *cm*. L'écartement entre ses axes est de 162 *cm*, comme montré dans la figure 3.1.

Les barres de guidage ont deux fonctions essentielles :
- le guidage du véhicule.

– l'alimentation en courant électrique des véhicules (la barre «positive» est alimentée en 750 V continu, et la barre «négative» est reliée à la terre).

Les barres positives sont recouvertes d'un capot de protection en matière plastique blanc principalement en partie aérienne. Ces barres sont disposées tout au long de la voie, sauf aux endroits des branchements ou des communications de voie.

Le support de ligne de transmission des automatismes est couvert par des de planches en matière plastique, et se situe dans le plan de roulement. Dans certaines zones le tapis principal est doublé par un tapis secondaire. Comme indiqué dans le chapitre précédent la voie se caractérise par un rayon de courbure horizontal minimal de 40 m, un rayon vertical minimal de 600 m et une pente maximale de 7%. Le dévers de la voie va de 0% à 13% avec une variation du dévers de $0,55\%/m$. Ces caractéristiques nous permettent d'utiliser l'hypothèse du monde plan.

Ce type de voie se différencie principalement des voies ferrées classiques par l'absence des traverses et l'existence des plusieurs structures linéaires et parallèles aux rails. Comme le montre la figure 3.2, les caractéristiques d'une voie ferrée maintiennent une continuité tout au long de la voie, comme l'espacement entre les traverses et le ballast et normalement l'espace est dégagé entre les deux voies. En revanche, dans le cas d'une voie de métro sur pneus automatisé, l'entre-voie est faite avec du béton, il existe d'autres structures comme les barres de guidages, le tapis, les passerelles, les boitiers de communications, etc.

(a) Voie ferrée.

(b) Voie pneumatique.

FIGURE 3.2 – Comparaison entre une voie ferrée classique et une voie pneumatique

3.2 État de l'art et approches testées

Pour la localisation des obstacles dans la trajectoire d'un train/métro, une des principales étapes nécessaires est l'extraction de la voie ferrée. Donc, en partant des connaissances de la détection routière dans le cas structuré, il est possible de citer différents types de méthodes à appliquer :

– La recherche de maxima et minima locaux du profil de niveau de gris [Lailler 1993].
– La transformée de Hough : détection d'ensembles de points situés sur une courbe paramétrique [Voisin *et al.* 2004].
– La transformation d'images : cette méthode transforme l'image par modification des coordonnées des pixels afin de corriger l'effet de la perspective [Broggi 1995].
– Les opérateurs de déclivité : Détection autoadaptative des contours.

Après avoir étudié plusieurs des hypothèses utilisées dans les différents systèmes de localisation et de suivi des marquages routiers, nous pouvons établir le tableau 3.1 indicatif de leurs avantages et inconvénients [Bertozzi *et al.* 2000].

Tableau 3.1 Comparatif des méthodes pour l'extraction de la voie.

Hypothèse	Avantages	Inconvénients
Région d'intérêt	On analyse qu'une petite portion de l'image, donc on a de grandes vitesses de traitement, en temps réel avec un faible coût hardware.	Le choix de la région d'intérêt est critique.
Largeur de la route fixe	Facilite le processus de recherche des voies. Robustesse dans les zones d'ombre ou les zones où les marques ne sont pas visibles	Problèmes quand la largeur/taille des marquages est variable.
Forme de la route	Modèle robuste qui facilite la reconstruction de la voie et le contrôle du véhicule.	Le modèle exige des équations complexes, pourtant une forte charge de calculs.
Connaissance *a priori* de la surface/inclinaison de la route	Cette approche simplifie les corrélations entre le monde réel et les pixels de l'image.	L'hypothèse n'est pas toujours applicable.

FIGURE 3.3 – Image représentative de la différence en texture d'une voie automobile et une voie ferrée
Dans ces images nous pouvons apprécier une binarisation des images avec un simple seuillage au niveau de gris pour une image routière et pour une image ferroviaire.

Afin de déterminer quelle part de l'expertise en vision accumulée dans le secteur routier est transférable à un système d'extraction des voies, il faut recenser les principaux points qui différencient les deux domaines. Une comparaison des scénarios est présenté dans le tableau 3.2 [Gschwandtner *et al.* 2010].

Tableau 3.2 Comparatif des méthodes pour l'extraction de la voie [Gschwandtner *et al.* 2010].

Route	**Voie ferrée**
La largeur de ligne est variable.	La largeur du rail est fixe.
La séparation entre les lignes est variable.	La séparation entre les rails est constante.
Variation du type de ligne.	Le rail est constant.
L'apparence générale est relativement homogène.	L'apparence générale n'est pas toujours homogène.
Les lignes sont faites pour être vues.	Les rails ne sont pas faits pour être vus.
Les lignes n'ont pas de volume, donc elles ne projettent pas d'ombres.	Les rails ont un volume donc ils projettent des ombres.
Le véhicule peut abandonner la voie.	Le véhicule aura tout le temps le même système de coordonnées.
La vitesse dépend des conditions météorologiques.	La vitesse est indépendante des conditions météorologiques.

En effet, l'écartement ou largeur des rails et leur taille sont fixes tout au long de la voie, par contre la largeur des lignes de la route dépend principalement du type de voie mais généralement elle est définie et limitée entre des valeurs maximales et minimales par des normes du trafic du pays. Également, la position relative de la voiture dans le centre de la voie est variable, ce qui rend la localisation des lignes plus complexe. En revanche, la position relative du train suit toujours les rails, et elle peut être déterminée en utilisant un modèle du train dépendant de sa largeur et de l'écartement de ses essieux.

Par ailleurs, les lignes de la route ont été conçues pour être vues d'une manière simple par l'œil humain, par contre les voies ferrées ont été construites pour guider les trains. La figure 3.3 montre un seuillage au niveau de gris et le résultat met en valeur la différence des textures existantes entre les deux images. Aussi, les voies ferrées sont proches les unes des autres et fusionnent dans les aiguillages, ce qui rend plus difficile l'extraction des «bons» rails. Le volume des voies ferrées peut former de l'ombre ce qui rend la détection plus difficile.

Ces caractéristiques spécifiques de la voie ferrée seront exploitées au maximum pour assurer la meilleure détection des rails dans l'image. Les différentes méthodes existantes dans l'état de l'art seront analysées par la suite pour finalement choisir celle qui s'adapte le mieux à notre environnement.

Aussi, parmi les méthodes possibles, celles basées sur les couleurs ou la texture de la zone à analyser ont été écartées puisque notre système doit être performant de nuit et comme de jour, en tunnel comme en aérien et sous différentes conditions météorologiques. Cette grande diversité de fonctionnement impose des contraintes fortes sur les techniques à appliquer.

La détection des rails, c'est un sujet qui a eu une grande importance dans les 10 dernières années [Beugin 2006]. Il existe différentes manières d'analyser le problème. Nous pouvons faire l'étude directement sur l'image avec la perspective captée, mais également faire une transformation de l'image en obtenant une vue depuis le haut, appelée «Inverse Perspective Mapping» aussi connue comme «Vue du Vol d'Oiseau» [Gangnet *et al.* 1984] de l'anglais «*Bird's eyes view*».

Le «filtre de Canny», ou détecteur de Canny [Canny 1986] est un filtre utilisé dans le traitement d'image pour la détection des contours. Il nous fournit de manière claire les contours avec une très bonne localisation. Cependant, il est nécessaire d'appliquer un seuillage au niveau de contours pour déterminer le niveau à partir duquel les variations d'intensité sont considérées des contours. Ainsi, un seuil trop bas conduira à la détection de faux positifs. Inversement, un seuil trop haut empêchera la détection de contours peu marqués, mais contenant de l'information utile.

Dans le cas en étude, à cause de la nature ferrée des rails et en raison de leurs polissage par les passages de trains, les rails ont des apparences très différentes suivant les conditions d'éclairement :

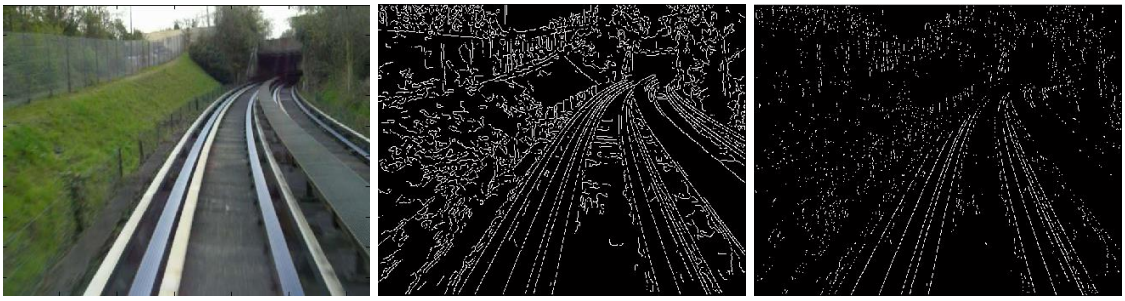
- brillants sur fond clair de jour par ciel dégagé
- gris sur gris avec peu de contraste de jour par ciel couvert
- noir sur gris de nuit s'ils sont éclairés par les phares du train (l'essentiel de la lumière émise par les phares est alors réfléchie dans la direction spéculaire, donc à l'opposé de la caméra).

Ceci peut rendre plus difficile la détection de rails que la détection des lignes routières.

(a) Image originale. (b) Image Canny avec un seuil bas. (c) Image Canny avec un seuil haut.

FIGURE 3.4 – Extraction des contours de la voie d'OrlyVal avec le filtre de Canny en fonction du seuillage appliqué.

Une autre possibilité est de diminuer le seuillage du filtre de Canny ce qui nous permettra de détecter la plupart des gradients présents dans l'image et *a posteriori* filtrer les contours en fonction de leur direction ou, autrement dit, de ne conserver que les contours dont la direction est compatible avec celle attendue pour les rails. Cette fourchette des angles pour le filtrage en fonction de l'orientation des gradients est définie par la projection en perspective des rails dans l'image. Comme montre la figure 3.4, le nombre de gradients est considérablement réduit.



(a) Image originale. (b) Image Canny avec un seuil très bas. (c) Image Canny filtrée en fonction de l'orientation des gradients.

FIGURE 3.5 – Extraction des contours de la voie d'OrlyVal avec un seuil bas avec le filtre de Canny et un filtrage *a posteriori* en fonction de l'orientation des gradients.

La détection des lignes dans une image est également possible grâce à la transformée de Hough [Ballard 1981], et a été déjà utilisée pour la détection des rails par [Rodriguez *et al.* 2012] dans ses travaux. Ils nous proposent d'analyser les rails par petites portions pour pouvoir les approximer à des segments de droites. Dans le domaine routier, [Tian *et al.* 2006] utilisent Hough, mais dans 5 parties de l'image pour pouvoir détecter la courbe approximative par des lignes droites. Cependant, le principal problème pour la détection des rails est la complexité de l'image, puisqu'il y a énormément des structures linéaires autour des rails comme montré dans la figure 3.7. En revanche, la connaissance *a priori* de la morphologie de la voie, facilite la distinction entre les lignes qui appartiennent aux rails et celles de la structure de la voie. Ainsi, [Rodriguez *et al.* 2012] propose d'utiliser la transformée de Hough pour détecter les rails.

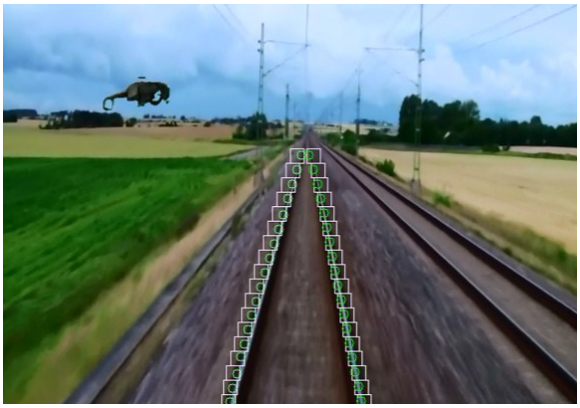


FIGURE 3.6 – Extraction des rails réalisée par transformée de Hough par portions. Image extraite de [Rodriguez *et al.* 2012]

Également, comme indiqué dans la section 2.4.3.1 l'orientation des lignes peut être utilisée pour les classifier dans l'espace de Hough comme le montre la figure 3.7 où deux couleurs ont été utilisées pour les différencier. Or, le problème de la transformée de Hough est qu'elle est lourde en calcul et implique l'utilisation des seuils au niveau de l'image de gradients et dans l'espace de Hough. Par conséquent, le résultat obtenu n'est pas toujours satisfaisant. Pour accélérer le processus de calcul [Rodriguez *et al.* 2012] propose de réduire la taille de l'image et pour améliorer l'extraction des contours une opération de fermeture morphologique est appliquée. Même si le temps de calcul se voit réduit, cette approche exige l'utilisation des seuils au niveau de la détection des gradients et au niveau de la recherche de lignes dans l'espace de Hough, ce qui peut limiter la détection des rails sous certaines conditions d'éclairage.

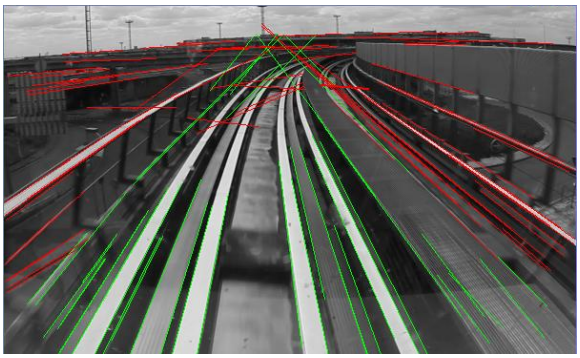


FIGURE 3.7 – Détection des lignes avec la transformée de Hough à partir de l'image binaire obtenue avec le filtre de Canny. Classification des lignes en fonction de leur orientation dans l'image pour déterminer si elles appartiennent ou pas à la voie

Par ailleurs, nous avons testé l'algorithme des cascades «*watershed*» [Beucher 1994] utilisé pour la segmentation des images à partir de la ligne de séparation des eaux (LPE), issue de la morphologie mathématique. Cet algorithme considère l'image en niveaux de gris comme un relief topographique, ce qui permet de trouver la ligne de séparation des régions de l'image. Le but de cet algorithme est d'éliminer la sur-segmentation causée par la LPE. Son fonctionnement consiste donc à supprimer les contours complètement entourés par des contours plus forts, en procédant à une fusion des bassins versants. Cette technique montre de très bons résultats, mais son temps d'exécution est difficilement compatible avec un traitement en temps réel.

De plus, les variations brutales de luminosité aux entrées et sorties des tunnels ajoutent une difficulté supplémentaire en raison du changement radical dans les niveaux de gris de l'image qu'elles engendrent. Deux exemples de segmentation de la voie avec *watershed* sont exposés dans la figure 3.8. Une solution possible serait d'ajouter des contraintes à la méthode *watershed*, par exemple trouver la ligne de séparation des eaux par sections dans l'image (opérateur de contraste local) puis réaliser la fusion des zones trouvées. Cependant, ajouter de la complexité à l'algorithme rendra le traitement en temps réel encore plus problématique.

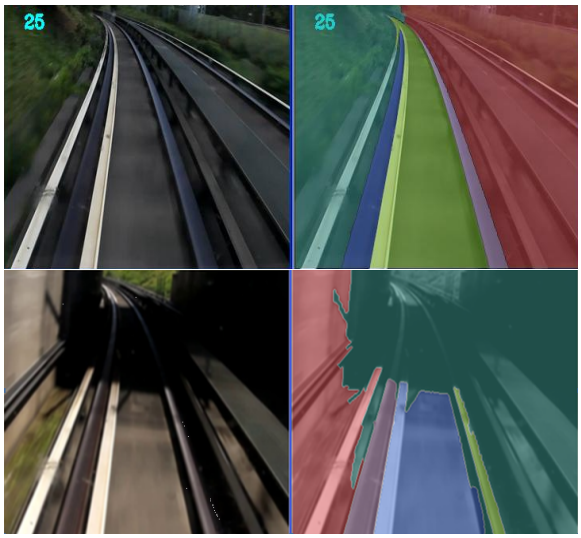


FIGURE 3.8 – Extraction des rails avec *watershed* et limitations de la technique.

La première approche originale développée et testée a été présentée dans ITSC 2012 (*International IEEE Conference on Intelligent Transportation Systems*), (voir annexe A.1).

L'idée principale était d'approximer les rails par des segments de lignes droites. Puisque la largeur de la voie est connue, nous pouvons analyser l'image avec 4 fenêtres glissantes le long des bords de chaque rail. Plus précisément, nous calculons les gradients transversaux dans chaque fenêtre, nous déterminons les positions de leurs maxima que nous utilisons pour faire glisser les fenêtres du bas de l'image vers le point de fuite.

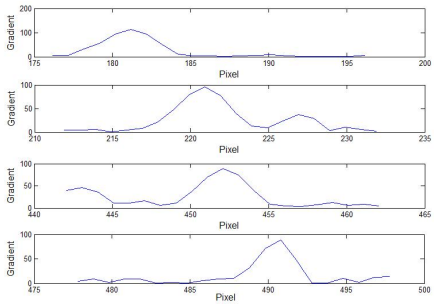
Pour contrôler ce déplacement, nous utilisons des estimations de position (modèle de ligne droite) et la position réelle (position des maximums). Grâce à la méthode des moindres carrés, nous actualisons le coefficient directeur des lignes droites et pouvons suivre les rails tout au long de l'image. Un exemple de fenêtres est exposé dans la figure 3.9a et les gradients obtenus à l'intérieur des fenêtres sont visibles dans la figure 3.9b.

Cette approche donne de bons résultats comme montré dans la figure 3.10. Cependant, elle se présente sensible aux occlusions, car c'est une approche d'analyse en continu. Si cette occlusion est supérieure à la longueur de la droite approximative, il existe un risque de perte de la direction des rails.

En outre, en raison des changements de pente, l'approximation de perspective faite pour placer les fenêtres ne convient plus et ces dernières peuvent être mal placées.

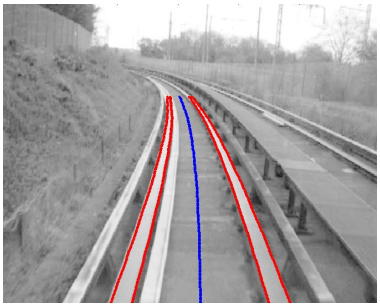
Quand cette hypothèse du monde plan $Z = 0$ n'est pas respectée, l'image en vue d'oiseau «*Bird's eyes view*» ou IPM de l'anglais *Inverse Perspective Mapping* se voit déformée. Une variation de Z introduit des erreurs dans la rétro-projection comme montre la figure 3.11. Il existe plusieurs méthodes pour corriger la limitation de l'hypothèse à partir du point de fuite [Nieto *et al.* 2007a] ou à partir de la variation de largeur des marqueurs routiers [Cerri & Grisleri 2005].

(a) Fenêtres glissantes.

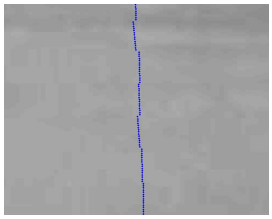


(b) Gradients à l'intérieur des fenêtres.

FIGURE 3.9 – Utilisation des fenêtres glissantes avec taille variable pour l'analyse des gradients.



(a) Résultat d'extraction des rails.



(b) Exemple des lignes droites suivies. Zoom de la partie centrale de l'image 3.10a

FIGURE 3.10 – Résultats obtenus avec la première approche.

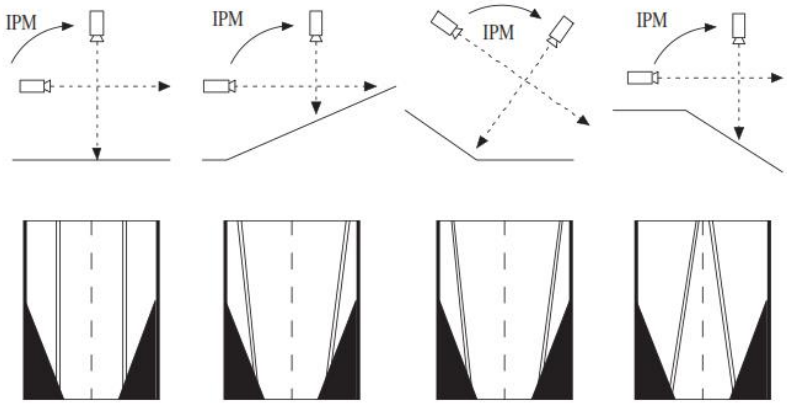


FIGURE 3.11 – Effets du non-respect de l'hypothèse du monde plan ($Z = 0$). Image extraite de [Nieto *et al.* 2007a].

Comme nous avons pu voir, les principaux problèmes soulevés pour la détection des rails sont dus à la présence des ombres qui font apparaître de forts contrastes (alternance ombre - soleil) entre la voie et les rails, d'une zone à l'autre de l'image. Ceci nous conduit à utiliser une approche locale pour la recherche des transitions maximum - minimum qui nous permettra de nous affranchir des problèmes de changement de contraste dans l'image.

De ce fait nous utilisons la technique par filtre adaptatif ligne par ligne présentée dans les travaux de [Bargeton 2009]. Le filtre de convolution de type $h(n) = [-1...-11...1]$ [Pomerleau 1995] (équation 3.1, où L représente la mi-largeur du filtre) permet de caractériser et mettre en valeur de façon efficace les lignes routières par un signal en forme de dents de scie. Tant que la largeur du filtre ne dépasse pas celle du rail, il y a création d'une dent de scie positive centrée sur un bord du rail suivi d'une dent de scie négative centrée sur l'autre bord.

$$h(n) = \begin{cases} \frac{1}{L}, & \forall n \in [1, L] \\ \frac{-1}{L}, & \forall n \in [-L+1, 0] \\ 0 & Reste \end{cases} \tag{3.1}$$

Pour minimiser le bruit, la taille du filtre doit être égale à deux fois la largeur du rail de façon que le minimum et le maximum du signal filtré correspondent aux 2 bords du rail. En revanche, lorsque la demi-largeur du filtre devient supérieure à la largeur du rail, les pics sont d'autant plus érodés que L augmente. Une fois les extremums détectés il faut les apparier.

(a) Sortie théorique du filtrage. (b) Image originale. (c) Image après convolution (d) Image des gradients détectés.

FIGURE 3.12 – Filtrage de convolution adaptatif pour la détection du marquage routière. Extraite de [Bargeton 2009]

Cependant, cette approche n'est pas adéquate pour notre application, car le plan de roulement du train n'est pas une surface homogène où les rails sont isolés. Dans notre cas, la présence des tapis à proximité des rails ne permet pas d'obtenir la caractérisation du rail par un signal dent de scie positif suivi d'un dent de scie négatif. Un exemple du résultat obtenu avec cette technique est montré dans la figure 3.13, où nous pouvons voir effectivement, comment les tapis ne permettent pas de mettre en valeur les rails.

Une autre approche est celle proposée dans les travaux de [Gschwandtner *et al.* 2010] réalisés pour la détection des rails. Il applique la transformée en vue vol d'oiseau ou IPM pour supprimer l'effet de la perspective et obtenir un aspect homogène des rails dans toute l'image. Cette transformée est simple à obtenir, mais reste lourde en calcul [Muad *et al.* 2004]. Un exemple est montré dans la figure 3.14.

Après avoir obtenu l'image IPM, [Gschwandtner *et al.* 2010] proposent d'utiliser un filtrage avec un masque DoG, de l'anglais *Difference of Gaussians*, pour détecter les rails dans l'image (3.2). L'équation 3.2 décrit le masque de ce type de filtre, où σ_1 et σ_2 représentent les écarts types des gaussiennes.

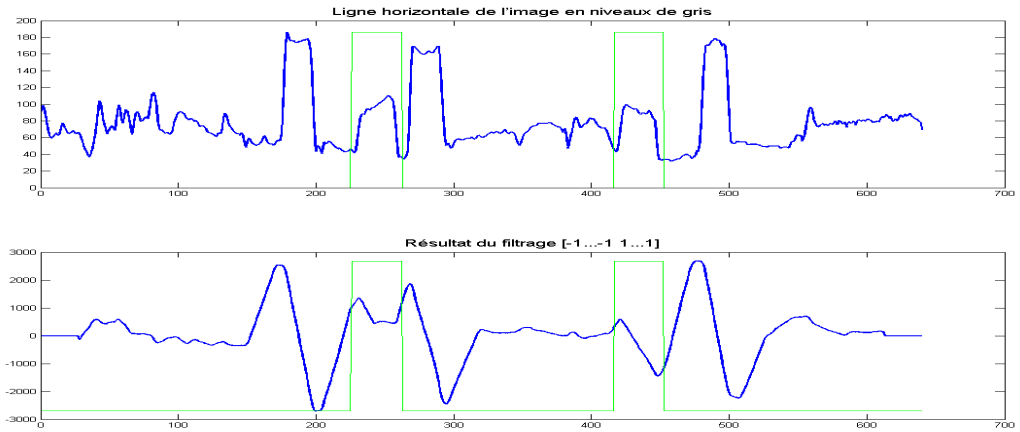


FIGURE 3.13 – Résultats de l'application du filtrage pour détecter les rails.
Cette image montre une ligne horizontale de l'image en niveaux de gris et le résultat de la convolution de cette ligne avec un filtre [-1 ... -1 1...1] de taille $L = 50$ (2 fois la largeur du rail dans l'image). Les lignes vertes nous indiquent la position des rails. Comme montre la figure, l'intensité des tapis est plus forte que celle des rails et en conséquence, les rails ne sont pas caractérisés par le signal *dent de scie*. Cependant, le résultat de la convolution met bien en valeur la présence d'autres structures comme les tapis et les rails de guidage

(a) Image de perspective (b) Vue d'oiseau ou IPM

FIGURE 3.14 – Transformation en vue d'oiseau (Image IPM)

$$h(n) = \frac{1}{\sigma_1\sqrt{2\pi}}\, e^{-\frac{(n-\mu)^2}{2\sigma_1^2}} - \frac{1}{\sigma_2\sqrt{2\pi}}\, e^{-\frac{(n-\mu)^2}{2\sigma_2^2}} \tag{3.2}$$

Nous pouvons utiliser cette même approche sans appliquer la transformation inverse, mais il faut tenir compte que la taille du filtre DoG doit s'adapter à la taille des rails dans l'image en perspective. Pour cella, les écarts types σ_1 et σ_2 des gaussiennes doivent varier.

Connaissant la taille des rails dans la partie basse de l'image en pixels (W), les coordonnées du point de fuite de l'image (u_{pf}, v_{pf}) et la taille de l'image $(Width, Height)$, on peut démontrer qu'une approximation de la largeur des rails dans l'image suit l'équation 3.3 [Bargeton 2009] :

$$w_v = W * \frac{(v - v_{pf})}{Height - v_{pf}} \tag{3.3}$$

où, w_v représente la taille en pixels des rails pour la coordonnée v de l'image de perspective.

Le masque du filtre DoG qui nous a donné les meilleurs résultats a des valeurs de $\sigma_1 = \frac{w_v}{4}$ et $\sigma_2 = \frac{w_v}{2}$. Pour déterminer la présence des rails dans le signal résultant du filtrage DoG, nous proposons de sélectionner seulement les maximums positifs supérieurs à un seuil et qui sont entourés des minimums négatifs. La figure 3.15 montre le résultat de l'application de l'opérateur DoG sur une ligne horizontale. De plus, dans la figure 3.16 nous pouvons voir en rouge superposé à l'image d'origine, un résultat de DoG. Cela nous permet de voir qu'effectivement les rails ont été bien détectés. Cependant, d'autres structures et des ombres ont été mises en valeur. Afin de réduire le nombre de points de contour détectés de façon erronée («faux positifs») nous proposons d'utiliser un critère de seuillage. La figure 3.15 montre en effet les jeux de lumières dans l'entrevoie, mis aussi en valeur mais avec une très faible amplitude.

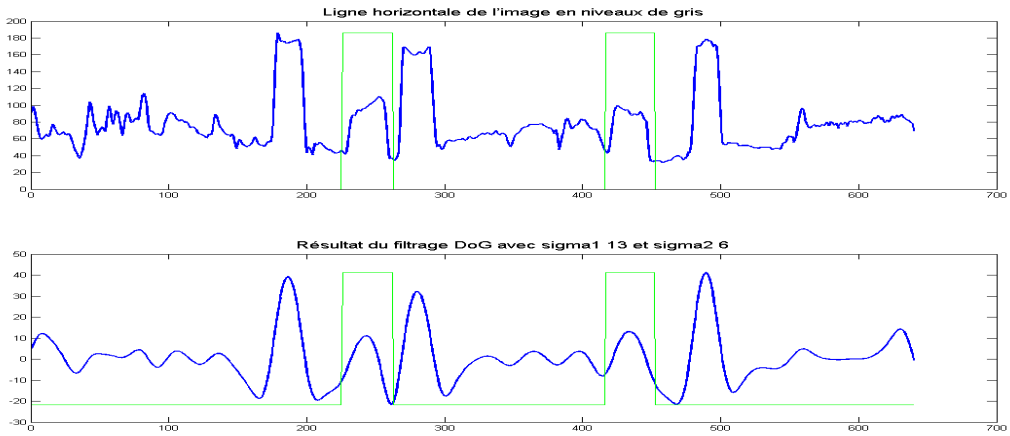


FIGURE 3.15 – Résultats de l'application du filtrage DoG pour détecter les rails.
L'utilisation d'un filtre DoG permet de mettre en valeur les rails et les autres structures de la voie car celles-ci ont tous une taille semblable. Par contre, des ombres dans l'image sont aussi également mises en valeur ce qui rend difficile la différenciation des rails des artefacts. Les pixels situés entre les valeurs 300 et 450 appartiennent à l'entrevoie et pour cette ligne par exemple, il y avait un jeu d'ombres.

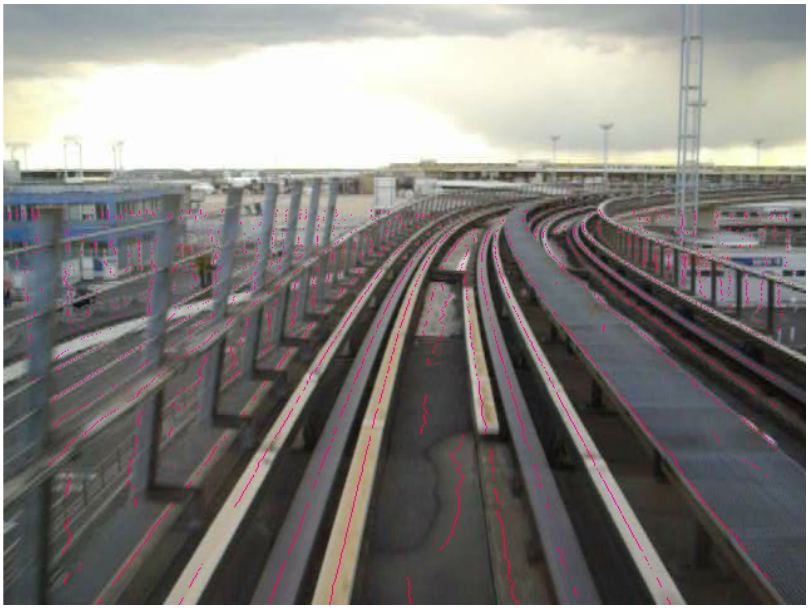


FIGURE 3.16 – Résultats de l'application du filtrage DoG pour détecter les rails.
Les points en rouge répresentent les maximuns locaux obtenus à partir de l'analyse du résultat du filtrage DoG avec un léger seuil. L'image montre que les rails sont bien extraits grâce à cette technique. Toutefois il y a beaucoup d'autres détections qui ne correspondent pas aux rails. Ces fausses détections ont une distribution linéaire, ce qui peut entrainer une fausse détection des rails dans les étapes suivantes.

3.3 Approche retenue

Après avoir étudié les différentes approches existantes, nous avons décidé de développer une technique basée sur l'analyse locale des gradients de l'image avec un seuil très bas pour satisfaire nos contraintes d'exploitation. Pour pouvoir réaliser l'analyse locale et accélérer le processus de calcul, nous proposons d'utiliser un masque pour définir une région d'intérêt et analyser les gradients dans l'image de perspective, sans obtenir une *Birds eye view* contrairement à [Gschwandtner *et al.* 2010]. Une fois l'analyse des gradients finie, seuls les points de contour avec des caractéristiques de type rail seront rétro-projetés pour obtenir leur position dans l'espace 3D. C'est dans cet espace que nous déterminerons la position des rails grâce à l'algorithme de RANSAC. RANSAC s'appliquera à chaque rail, droit et gauche. Une approximation par un polynôme de degré 2 est utilisé pour chaque RANSAC. Une fois retenus les meilleurs points de contours, ces derniers seront translatés au centre de la voie pour obtenir la fonction qui définit la courbure de la voie. Puis un filtrage temporel peut être utilisé pour réduire le nombre des itérations de RANSAC pour les images suivantes. Finalement, à partir des données obtenues, nous déterminerons la distance maximale de détection des rails.

Les étapes de notre approche sont :
- Sélection du Masque.
- Analyse des gradients : Détection des points de contour.
- Rétro-projection des points de contour 2D à 3D.
- Obtention du polynôme de la voie : RANSAC
- Filtrage temporel.
- Détermination de la distance maximale de détection.

3.3.1 Sélection du Masque

Une région d'intérêt (ROI de l'anglais *Region Of Interest*) est une région de l'image qui est analysée avec une plus grande précision, en général parce que cette région présente un intérêt

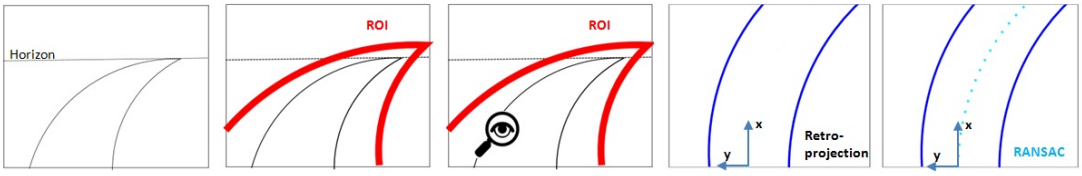


FIGURE 3.17 – Phases d'extraction des rails

particulier. Dans notre cas, elle définit l'emplacement de la voie dans l'image. Cette plus grande précision se fait au détriment des autres zones de l'image qui ne sont pas analysées avec l'objectif de réduire le nombre des points inspectés et d'accélérer le processus de détection de la voie. La sélection de la ROI est faite grâce à la connaissance *a priori* de la géométrie de la voie et au fur et à mesure que la voie est détectée cette ROI suivra la position de la voie.

Notre ROI a été définie comme une forme de trapèze avec son sommet dans la position du point de fuite des rails et une base plus large que la largeur de la voie comme montre la figure 3.18.

FIGURE 3.18 – Représentation du masque de la ROI.

Dans une première étape, ce point de fuite est positionné sur la ligne de l'horizon et est centré dans l'image. Une fois que les rails ont été détectés, le nouveau point de fuite correspondra au point de fuite des rails. Ce suivi du point de fuite des rails nous permettra de bien positionner la ROI dans l'image car le choix de son emplacement est un aspect critique.

Pour pouvoir connaitre la position du point de fuite des rails, c'est-à-dire, le point dans l'image où les rails fusionnent, nous proposons d'utiliser la tangente aux rails à la distance maximale détectée. Autrement dit, une fois que l'équation de la parabole centrale définie par la position des rails a été obtenue, nous extrapolons la tangente aux rails à l'infini pour obtenir le point de fuite des rails.

Pour obtenir la valeur de la tangente, le théorème suivant a été appliqué :

Soit $f(x)$ la fonction réelle définie sur l'intervalle I et soit la distance maximale a, un réel de I tel que f soit dérivable en a, de nombre dérivé $L = \dot{f}(a)$. On note C la courbe représentative de f au voisinage de A $(a; f(a))$. L'équation réduite de la tangente T à C au point d'abscisse a est :

$$y = \dot{f}(a) \times (x - a) + f(a) \tag{3.4}$$

Cette méthode est valable puisque les voies ferrées font rarement un demi-tour et aussi parce que l'approximation de la voie utilisée est un polynôme du 2ème dégrée. Plusieurs travaux dans le

domaine routier se basent aussi dans cette approximation [Chen & Wang 2006] et [Choi *et al.* 2012]. Il pourrait être envisageable d'obtenir le rayon de courbure à la distance maximale de détection et d'utiliser l'équation du cercle pour prolonger les rails et ensuite, obtenir la position du point de fuite recherché, mais il semble suffisant l'effet d'utiliser la tangente. La figure 3.19 montre des résultats de cette méthode dans le plan image et dans l'espace $3D$.

3.3.2 Analyse des gradients

L'approche proposée pour la détection des rails consiste à analyser les gradients horizontaux de l'image. Dans cette analyse nous cherchons une variation des niveaux de gris dans l'image avec une largeur similaire à la largeur du rail et ces gradients doivent avoir une orientation similaire. Nous avons pu voir dans la figure 3.5 comme était possible d'utiliser un filtrage angulaire pour mieux cibler le seuillage de Canny. Plus précisément, nous cherchons dans l'image des gradients écartés de la largeur du rail et qui ont une orientation similaire comme exposé dans le pseudo-code 3.3.1. Le résultat de notre approche est représenté dans les figures 3.20 et 3.21, pour une ligne horizontale de l'image et pour toute l'image respectivement.

Cependant, comme expliqué dans la section 3.2, l'analyse des gradients dans l'image reste une tâche cruciale dans la recherche des rails dans l'image. De plus, la présence du tapis de transmission blanc rend difficile l'utilisation de seuils pour déterminer les gradients dans l'image.

Algorithm 3.3.1: APPROCHE PROPOSÉE()

```
GH comment: Obtention des gradients horizontals de l'image
G comment: Obtention de l'orientation des gradients de l'image
for i ← IMAGE_HEIGHT to HORIZONT_LINE
do { f = GH(i, :)
     x = 1
     while x < IMAGE_WIDTH
     do { if f(x) is maximum & f(x) > SEUIL
          then { x2 = x + 1
                 while x2 < IMAGE_WIDTH & x2 < (x + taille_rail + marge)
                 do { if f(x2) is mimimum & f(x2) < −SEUIL &
                      G(i, x) et G(i, x2) sont similaires
                      then { Point de contour = (x + x2)/2
                             comment: Point de contour detecté
                      x2 = x2 + 1
          x = x + 1
```

Pour mieux évaluer l'efficacité de cette technique, nous proposons de confronter la pertinence la méthode proposée et la technique DoG à la vérité terrain. L'exemple de vérité terrain exposé dans la figure 3.22 est composé de deux rails, deux tapis et deux rails de guidage. Étant donné que les deux techniques nous donnent pour résultat le centre des contours considérés comme un possible rail, c'est le nombre de points obtenus que nous avons analysé. Les points de contour qui correspondent correctement à la position des rails ou tapis sont considérés comme «vrai-positifs», les autres sont considérés «faux-positifs». Puisque le but est de caractériser les rails et les tapis, plus nous avons de points disponibles, plus précis sera le résultat. En revanche, un nombre important de «faux-positifs» peut nous amener à des résultats inexacts.

Puisque les deux techniques exigent d'utiliser un seuil pour réduire le bruit, nous proposons d'utiliser le même critère pour le déterminer. Nous avons défini que seules les variations de niveau de gris supérieures à 8 seront prises en compte. Cela veut dire que les très faibles variations ne

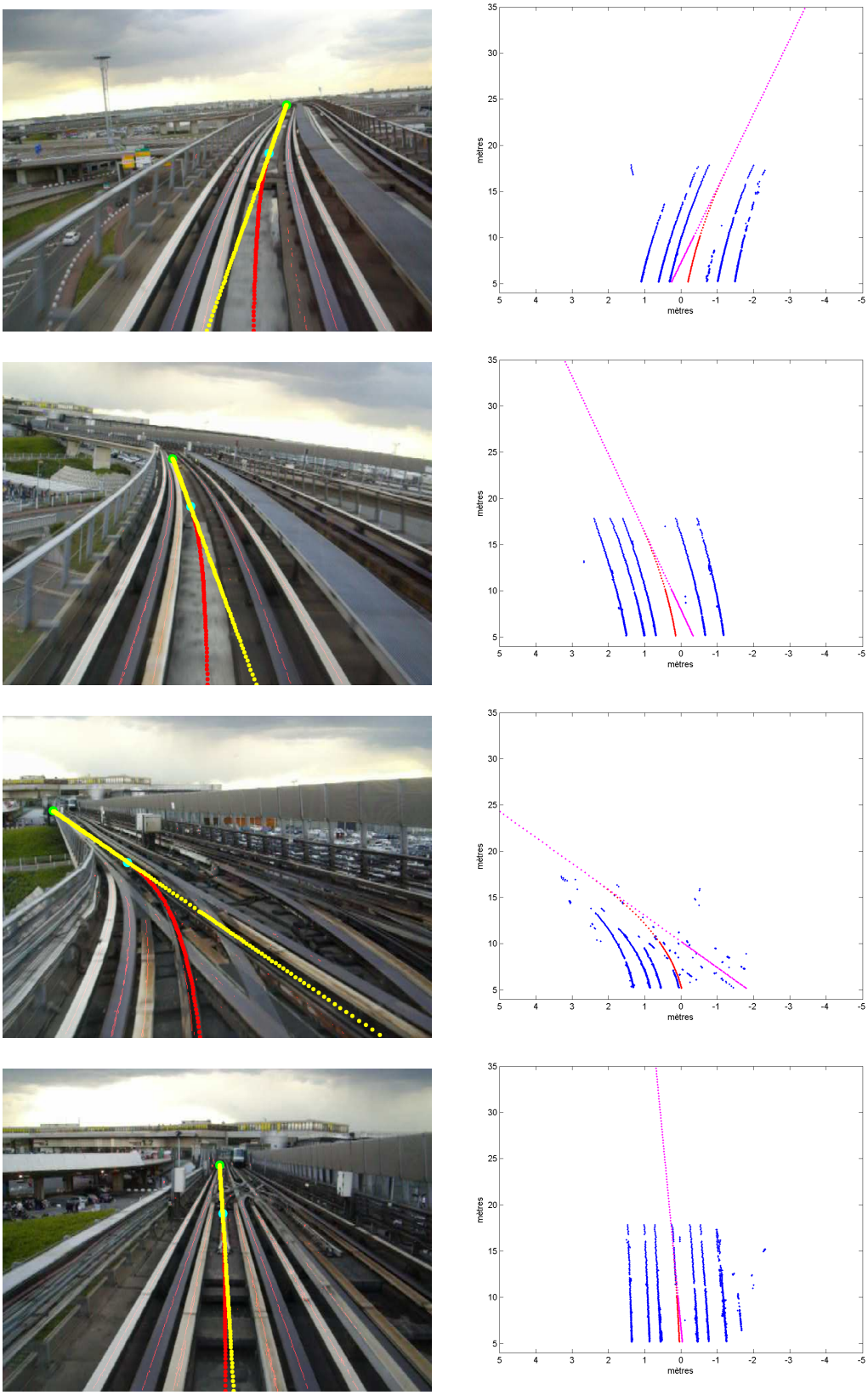


FIGURE 3.19 – Résultats d'interpolation du point de fuite des rails.
Dans les images de gauche, nous pouvons observer la parabole détectée en rouge, la distance maximale de détection représentée par un point de couleur cyan et la tangente à ce point extrapolée à l'infini en jaune. Les images de la colonne droite représentent la position des pixels dans le plan $3D$, où les points bleus sont les gradients déterminés comme candidats à appartenir à la voie, les points rouges représentent la courbe obtenue et les points magenta la tangente de cette courbe à la distance maximale de détection.

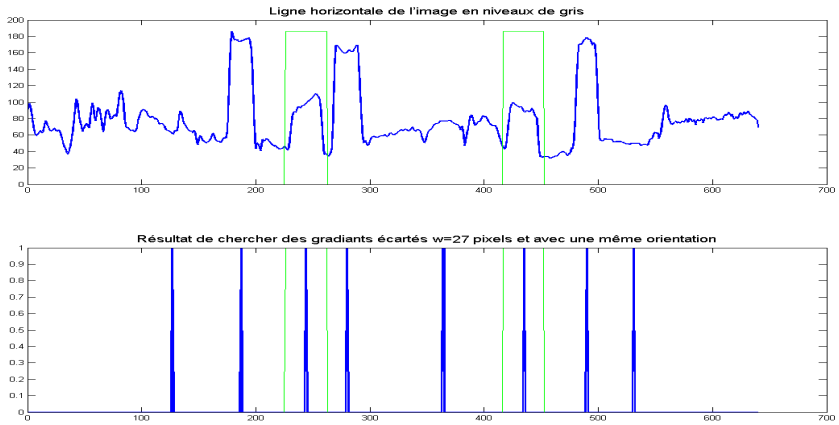


FIGURE 3.20 – Résultats de l'analyse des gradients proposée pour détecter les rails.
L'utilisation de la méthode proposée permet de mettre en valeur les rails et les autres structures de la voie. Le signal de sortie vaut 1 pour indiquer le centre d'un gradient de largeur w qui a dans ses extrémités deux angles d'orientation similaires. Comme pour la méthode précédente, certaines ombres de l'image ont été extraits de façon erronée. Si les rails sont plus sombres que l'environnement(la nuit les rails ne reflètent pas la lumière vers la caméra) il suffit d'inverser le SEUIL, pour trouver les zones plus sombres.

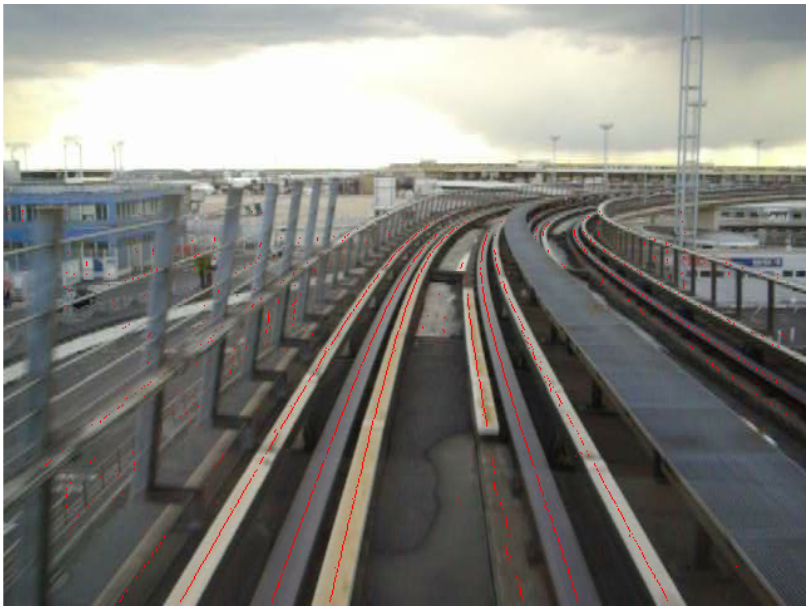
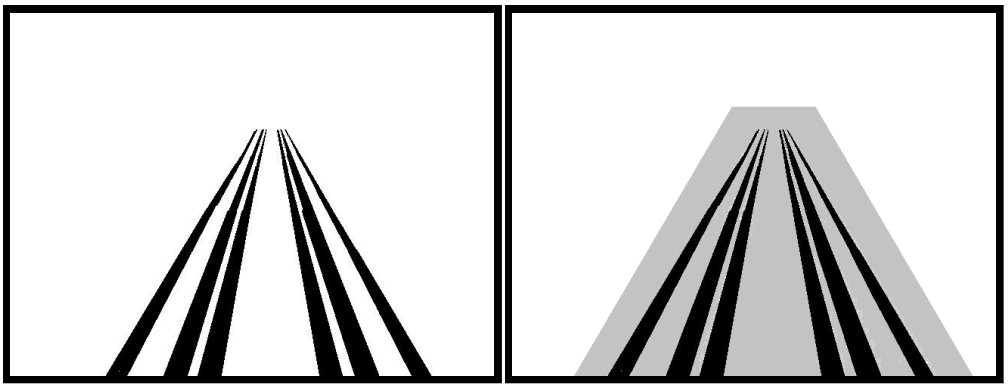


FIGURE 3.21 – Résultats de l'analyse des gradients proposée pour détecter les rails.
Dans cet image, les points en rouge représentent le centre des points de contours obtenus à partir de l'analyse des gradients proposée sans avoir appliqué aucun seuil. L'image montre que les rails sont bien extraits grâce à cette technique. Cependant, il y a d'autres détections qui ne correspondent pas aux rails. Nous proposons une méthode de filtrage *a posteriori* de ses détections erronées.

seront pas retenues. Cette différence de 8 en niveaux de gris, peut être considérée comme un seuil très faible car elle représente une variation de 3.125% dans une image de 256 bits.



(a) Vérité terrain de la position des rails dans l'image.

(b) Vérité terrain et le masque appliqué pour la recherche des gradients.

FIGURE 3.22 – Vérité terrain de l'image à analyser et le masque appliqué.
En considérant un point de contour par rail et par tapis pour chaque ligne horizontale de l'image, le nombre total de points de cette vérité terrain est de 1534 points. La taille de l'image est de 480×640 pixels. De ce fait, les rails et les tapis occupent un 8.5% de l'image, depuis la ligne 480 jusqu'à la ligne de l'horizon 198. Le masque réduit la zone de recherche à un 40% de l'image.

Une comparaison des résultats des deux techniques en utilisant le pourcentage de nombre de points détectés dans l'image par rapport au nombre des points dans la vérité terrain est exposée dans le tableau 3.3. Pour cela, une section de 100 images en ligne droite a été choisie.

Nous comparons le nombre des points de contours détectés et le nombre de points erronés. Pour examiner la robustesse des deux techniques vis à vis du bruit, nous avons inspecté toute l'image, ce qui nous permet aussi de vérifier l'efficacité du masque. Nous avons décomposé les fausses détections en deux, dans l'image complète d'une part, à l'intérieur du masque d'autre part.

Dans la figure 3.23 nous pouvons voir la variation des taux de détection pour les 100 images sélectionnées. Dans le tableau 3.3 nous trouvons les valeurs moyennées sur l'ensemble des images.

Tableau 3.3 Valeurs moyens de taux de détections des points de contour pour 100 images.

	Vrai positifs	Total de Faux positifs	Faux positifs avec masque
DoG	61.21 %	145.5 %	44.65 %
Analyse de Gradients	74.01 %	58.42 %	19.19 %

À la vue des figures 3.16, 3.21 et 3.23 et du tableau 3.3, nous considérons que le filtrage *DoG*, même s'il offre de très bons résultats, souffre d'un grand nombre des faux-positifs. De plus, ces fausses détections ont des formes très linéaires ce qui peux nous induire à les considérer comme des rails. En fin, cette technique ne nous permet pas de garder l'information de l'orientation des gradients dans l'image, ce qui comme montré dans la figure 3.5 aide à discriminer les différents contours irréguliers présents dans l'image et explique pourquoi la technique proposée montre des meilleurs résultats.

En résumé, la technique proposée est basée dans la recherche des gradients dans l'image, supérieurs à un seuil faible, écartés d'une distance «w» et avec une orientation semblable.

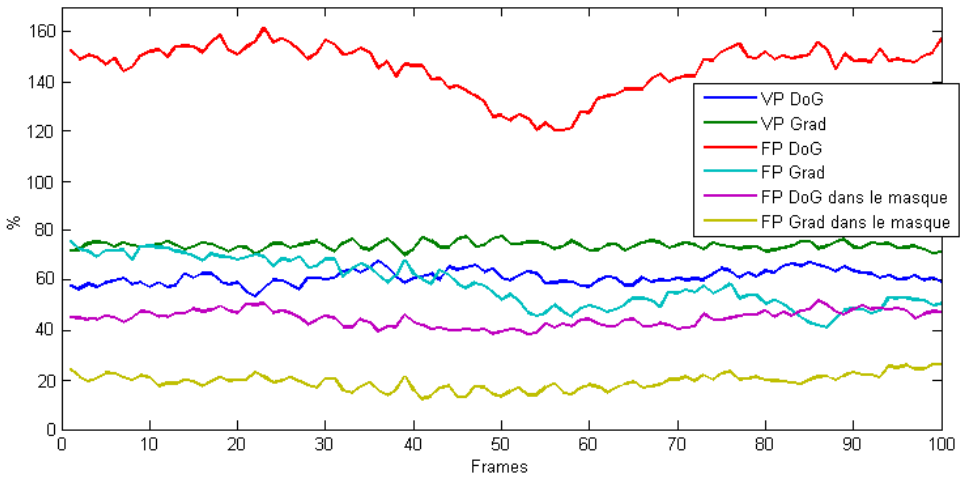


FIGURE 3.23 – Comparatif du nombre de «VP : vrai-positifs» et «FP : faux-positifs» pour les techniques de différence des gaussiennes (DoG) et la méthode proposée.

3.3.3 Recherche du modèle dans l'espace 3D

Une fois que les points de contours sont déterminés, le traitement principal consiste à les mettre en correspondance avec un modèle prédéfini décrivant la courbure de la voie ferrée. Le modèle choisi est un polynôme de 2ème dégrée [Kaske *et al.* 1997], [Jung & Kelber 2004] et [Choi *et al.* 2012]. On peut considérer que ce polynôme est suffisant pour caractériser la courbure de la voie dans les limites du champ de vision de la caméra dû aux contraintes de rayon de courbure existant dans les lignes VAL.

Pour bien exploiter les contraintes de la voie, (largeur, rayon de courbure minimum, etc) nous transformons les points des contours obtenus en des points $3D$ situés dans le monde réel grâce à l'application de la rétro-projection obtenue à partir du modèle complet sans distorsion (équation 2.2). D'autres techniques transforment complètement l'image pour obtenir une transformée inverse complète, mais en transformant seulement les points de contour nous avons un gain du temps de calcul très important. L'hypothèse du monde plan est retenue pour pouvoir réaliser cette rétro-projection. L'impact de cette hypothèse en présence des pentes est analysé dans la section 3.3.5.1.

C'est dans ce nuage des points $3D$ que nous allons chercher les coefficients du modèle de la voie.

Pour ajuster les points de contours au modèle, nous avons choisi d'appliquer la méthode de vote probabiliste de RANSAC (*RANdom SAmple Consensus*)[Fischler & Bolles 1981], qui est couramment utilisée pour des ajustements robustes à des modèles en présence de bruit.

En outre, cette méthode a été conçue afin de réduire le temps de calcul des méthodes de votes classiques comme la transformée de Hough.

Les étapes de cet algorithme pour trouver un polynôme de degré n sont les suivantes :
- On choisit aléatoirement $n+1$ points parmi l'ensemble.
- On calcule les paramètres du polynôme passant par ces points.
- Par calcul de la distance des autres points, on détermine les «points proches» du modèle (*inliers* de l'anglais) ayant une distance inférieure à un seuil de tolérance prédéfini.
- Si le nombre d'*inliers* est suffisant, alors on calcule l'erreur par rapport à tout l'ensemble des points.
- Si l'erreur totale du modèle est inférieure à l'erreur du meilleur modèle précédent, ce modèle-ci est assigné comme le nouveau meilleur modèle et l'ensemble des points proches (*inliers*) est

gardé (*consensus points*).
- On recommence la procédure avec un nouvel ensemble de points jusqu'à un nombre maximum de tentatives.

3.3.3.1 Nombre d'itérations de RANSAC

Le nombre minimal d'itérations nécessaires pour que le «meilleur modèle» ne contienne que des "*inliners*" au seuil de probabilité P donné peut être déterminé de façon théorique. Si on désigne par W la probabilité de choisir à chaque itération un *inliner* valable et que les n points nécessaires à définir la fonction soient sélectionnés de façon indépendante, dans ce cas, la probabilité pour que les n points soient tous des *inliners* est de W^n, et $(1-W^n)$ es la probabilité qu'au moins un de ces points ne fasse pas partie des *inliners* à chaque tirage. La probabilité de reproduire un tel tirage imparfait K fois est donnée par l'équation 3.5. Elle correspond à la probabilité pour que l'algorithme ne sélectionne aucun échantillon de n *inliners* à la fois. Elle est alors égale à $(1-P)$, d'où l'égalité suivante :

$$(1-W^n)^K = 1-P \tag{3.5}$$

où

$$W = \frac{Nombre\ de\ points\ valables}{Nombre\ total\ de\ points} \tag{3.6}$$

Dans notre cas, il suffit de deux points puisque nous cherchons à ajuster des paraboles passant par un point déjà connu : celui qui correspond à la position du rail dans la partie base de l'image transformée dans ces valeurs $3D$.

Nous fixons également une probabilité $P = 0.999999$ que dans K^n itérations les coefficients de la courbe seront détectés.

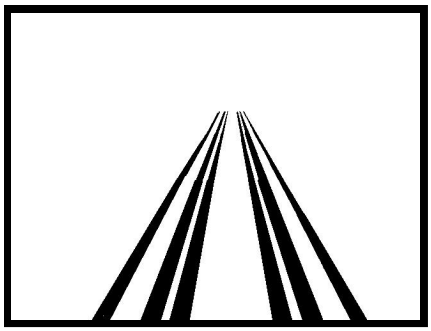
Pour déterminer W, considérons le cas plus compliqué consiste à trouver la parabole d'un rail (droit ou gauche) en présence de tous les éléments de la voie, à savoir, les deux pistes de roulement, les deux tapis et les rails de guidage, tous présents dans l'image depuis la partie basse jusqu'au point de fuite, comme montré dans la figure 3.24. Dans ce cas, chaque élément de la voie apportera le même nombre de points M à l'ensemble des points $3D$ et la probabilité W est définie par 3.6. Finalement, en substituant dans 3.6 et 3.5, la probabilité $W = 1/6$ et le nombre des itérations K optimal résulte de 490 (voir 3.7 et 3.8).

$$W = \frac{M}{6 \cdot M} = \frac{1}{6} \tag{3.7}$$

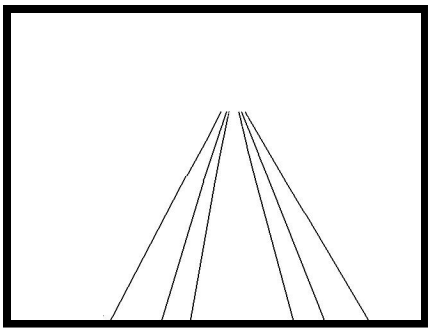
$$K = \frac{\log(1-P)}{\log(1-W^n)} = \frac{\log(1-0.999999)}{\log(1-\frac{1}{6}^2)} \approx 490 \tag{3.8}$$

Avec cette considération idéale, le nombre d'itérations pour obtenir le modèle ne dépend pas de la taille de l'image ni de l'inclinaison de la caméra. Si maintenant nous prenons en compte, les points de contour erronés, ceux qui n'appartiennent pas à la voie, soit environ un 20% du nombre des points de la vérité terrain (tableau 3.3), la nouvelle probabilité W est de $\frac{1}{7.2}$ et le nombre d'itérations passe à $K \approx 710$.

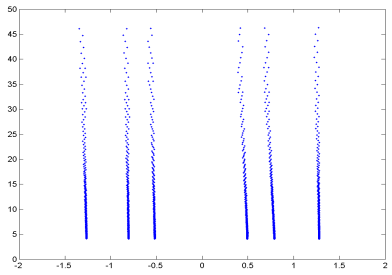
(a) Image originale.



(b) Vérité terrain.



(c) Détection des gradients dans la vérité terrain.



(d) Points dans le repère 3D.

FIGURE 3.24 – Vérité terrain et sa rétro-projection.

3.3.3.2 Score du modèle

Dans l'algorithme de RANSAC, à chaque estimation un jeu aléatoire de données est testé et il lui correspond un score pour les paramètres obtenus. Le jeu de paramètres ayant le score le plus élevé est retenu comme le résultat de l'estimation. Normalement, le vote est basé sur le nombre de points dont la distance au modèle est inférieure à un seuil donné (une tolérance minimale).

Comme ce critère peut donner des faux résultats si nous prenons de façon indépendante chaque rail, nous proposons que le test d'appartenance prenne en compte l'existence de tous les éléments de la voie. Cela nous permettra de réduire le nombre des fausses détections, car le meilleur modèle satisfera l'ensemble des points du nuage obtenu.

Autrement dit, à chaque itération, une fois déterminé un modèle nous calculons la position théorique des autres structures par rapport au modèle (le rail opposé, les tapis et les rails de guidage). Nous proposons d'utiliser une fonction gaussienne pour pondérer l'erreur des modèles itératifs de l'algorithme de RANSAC pour chaque structure de la voie. Ces fonctions sont caractérisées par une densité de probabilité suivant la loi normale, centrée sur la position de chaque structure.

Il faut également tenir compte du fait que la résolution spatiale après une rétro-projection varie en fonction de la distance, c'est-à-dire que pour les points proches (ceux de la partie basse de l'image), la résolution spatiale est bonne (quelques mm entre 2 pixels), et plus on monte dans l'image vers le point de fuite, plus grand est l'écart en distance de deux pixels contigus.

De ce fait, nous proposons que les gaussiennes aient un écart type variant en fonction de la distance. Cet écart type sera plus petit pour les points proches et plus large pour les points plus lointains, mais sans jamais superposer complètement deux gaussiennes adjacentes.

Les valeurs d'écart type ont été déterminées *a priori* de façon linéaire à partir de la coupe

transversale de la voie (figure 3.1) où la distance minimale entre les rails et les tapis est de $0,255\ m$. La largeur des gaussiennes doit être inférieure à cet écartement pour bien séparer chaque structure et avoir un résultat du modèle plus précis. La variation de l'écart type prend en compte ces distances minimale et maximale de détection des rails, et il est plus élevé en fonction de la distance, sans jamais associer deux gaussiennes voisines.

La figure 3.25 montre l'exemple des 6 gaussiennes qui satisfont le mieux les points $3D$ de la figure 3.24.

Par conséquent, la fonction de score pour une gaussienne est définie par l'équation 3.9, et le score total du modèle sera la somme des scores pour chacune des gaussiennes calculées en fonction de la présence ou pas de la structure dans l'image.

$$Score(M_k) = \sum_{i=1}^{N} \frac{1}{\sigma(d_i)\sqrt{2\pi}} \exp - \frac{(p_i - (M_k(p_i) + S_j))^2}{2(\sigma(d_i))^2} \tag{3.9}$$

où :

- M_k est le modèle k de la série de K modèles aléatoires du RANSAC,
- N : est le nombre total de points $3D$,
- p_i : est un point i du nuage de points,
- S_j est la distance entre le modèle et la structure j, où $j \in [1, S]$ et S est le nombre de structures prises en compte.
- $\sigma(d_i)$: représente l'écart type de la gaussienne en fonction de la distance du point p_i

$$ScoreTotal(M_k) = \sum_{i=1}^{S} Score(M_k)_i \tag{3.10}$$

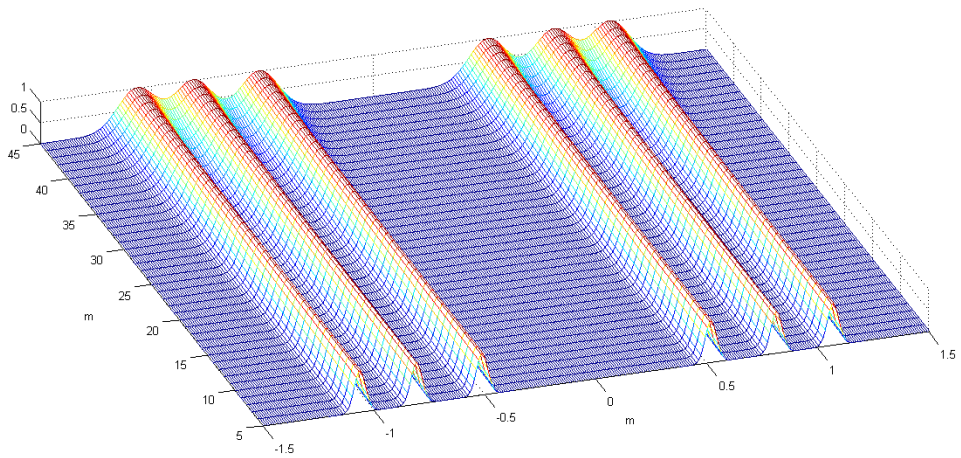


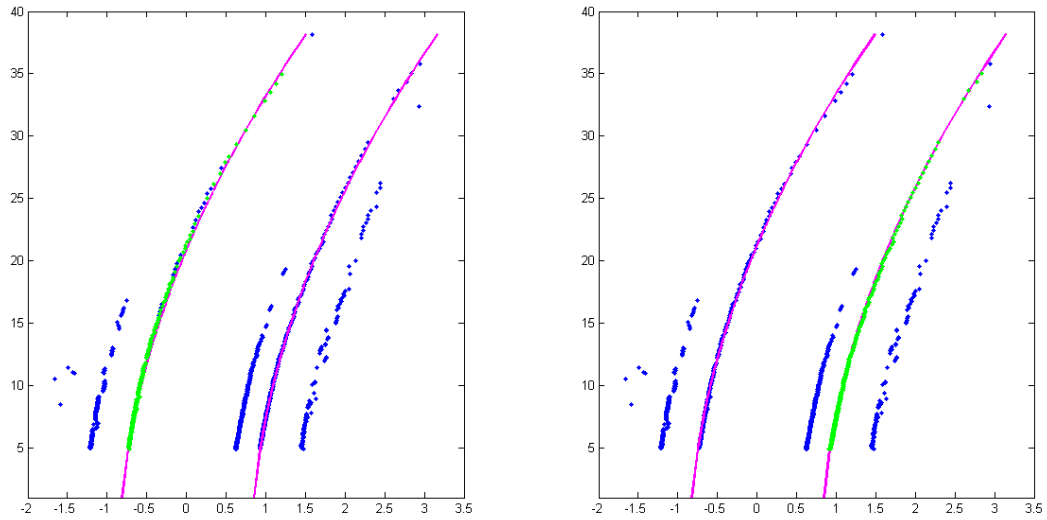
FIGURE 3.25 – .
Répresentation des 6 gaussiennes avec un écart type en fonction de la distance.

De ce fait, dans le processus itératif le meilleur modèle sera celui qui a un score plus élevé.

3.3.3.3 RANSAC pour chaque rail

Comme indiqué précédemment, les variations de changement de pente dans le milieu ferroviaire sont limitées, ce qui fait qu'on ne trouvera pas de cas comme les exemples de la figure 3.11. Cependant, des changements de pente même faibles vont introduire des déformations dans les nuages de points obtenus. Pour surmonter cette difficulté, nous proposons d'obtenir un modèle pour chaque rail et de réaliser ensuite une translation vers le centre de la voie des meilleurs points retenus.

Les étapes à suivre sont : RANSAC pour chaque rail en tenant en compte de la présence de toutes les structures ; la translation des meilleurs points (les *inliers*) vers le centre de la voie, et finalement une régression des moindres carrés pour obtenir la fonction de la courbe qui définit les rails. Un exemple du résultat obtenu avec RANSAC pour obtenir les rails de droite et de gauche est montré dans la figure 3.26.



(a) RANSAC pour le rail de gauche.

(b) RANSAC pour le rail de droite.

FIGURE 3.26 – RANSAC individuels pour chaque rail.

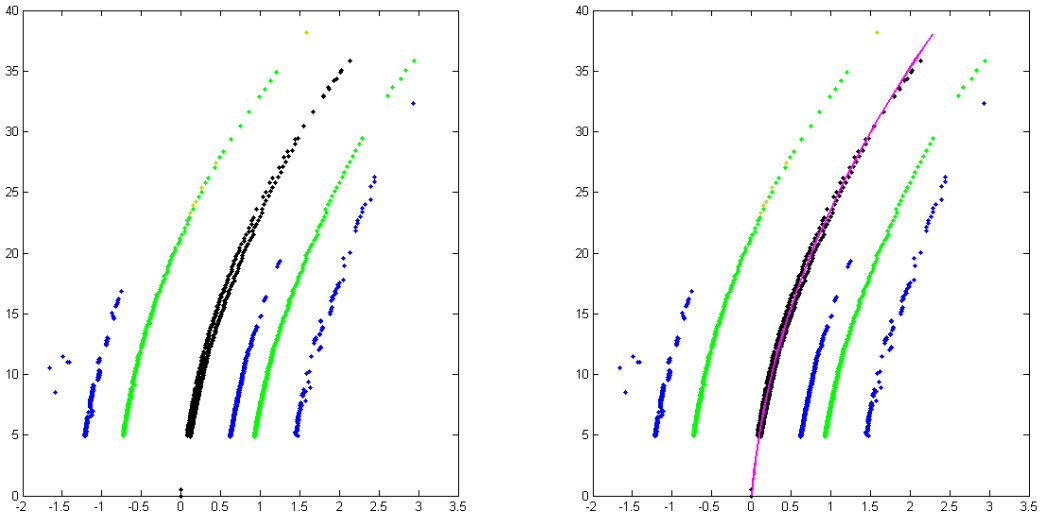
Dans ces deux images nous pouvons voir en bleu les points de contours representés avec les coordonées $3D$. En vert, les points considérés appartenant au modèle trouvé (les *inliers*). Les traits magenta représentent les centres des deux gaussiennes utilisées pour trouver le modèle.

Une fois obtenu chaque rail, les *inliers* de chaque modèle sont déplacés vers le centre de la voie l'écartement des rails avec l'objectif d'obtenir l'équation de centrale de la parabole cherchée. Pour cela, une régression des moindres carrés est réalisée dans le nouveau ensemble de points. Cette régression aura la particularité d'être pondérée en fonction de la distance des points. Plus le point est loin, plus faible est son d'influence dans le résultat de la courbe. L'objectif est de réduire les erreurs et de donner plus d'importance aux points proches. Ce poids W_i est défini dans l'équation 3.11.

$$WSS = \sum_{i=1}^{n} W_i r_i^2 \quad , W_i = \frac{1}{D_i}, \quad r_i^2 = y_i - f(x_i, \beta) \tag{3.11}$$

où n représente le nombre total de points retenus par RANSAC et D_i la distance au repère $O_W = (0, 0, 0)$ du point p_i.

Un exemple de la translation des *inliers* de RANSAC pour chaque rail et l'obtention de la parabole centrale est exposé dans la figure 3.27. Cette étape finale de régression par moindres carrés



(a) Consensus des points retenus avec RANSAC avec tous les rails déplacés au centre de la voie.

(b) Parabole obtenue avec régression des moindres carrés pondérée..

FIGURE 3.27 – Régression des moindres carrés des résultats de RANSAC.

Les *inliers* retenus par RANSAC pour chaque rail sont représentés en vert dans l'image 3.27a et 3.27b. Ces points sont déplacés vers le centre de la voie une distance égale à la moitié de la largeur de la voie. Une fois déplacés (points noirs dans les deux images), la parabole que caractérise la courbure de la voie est définie à partir d'une régression de moindres carrés pondérée (représentée en magenta dans l'image 3.27b).

nous permet de nous affranchir des problèmes des changements faibles de pentes.

3.3.4 Dépendance temporelle

Puisque nous cherchons à réaliser un système temps réel, deux RANSACs de 710 itérations chacun, à une cadence de 25 images par seconde ne semble pas l'option optimale. Par conséquent, nous avons optimisé la recherche des rails en ajoutant de la dépendance temporelle. Puisque les trains sont guidés il est facile d'introduire cette causalité en connaissant le modèle du train. Comme montré dans la figure 2.5 et l'équation 2.1 l'angle entre la caisse et l'axe de la voie «le lacet ψ_{caisse}» peut être obtenu si on connaît à tout instant les positions avant et arrière de la caisse (vecteurs t_1 et t_2).

Cette compréhension nous permet d'obtenir une relation entre l'axe de la voie et l'axe d'orientation de la caméra pour réaliser la rétro-projection des points de contours détectés dans un repère fixe, dans lequel nous pouvons introduire la dépendance temporelle.

Si nous arrivons à augmenter la probabilité W du tirage au sort, le nombre d'itérations se verra réduit significativement. Par exemple, si maintenant $W = 0.5$ et la probabilité de trouver les coefficients $P = 0.999999$ reste la même, le nombre total d'itérations de RANSAC descend à 48. Ceci permettra d'augmenter considérablement la vitesse d'exécution de l'algorithme.

La solution apportée au problème cité ci-dessus consiste à séparer les points candidats à appartenir au modèle de l'ensemble des points obtenus avec la rétro-projection. Cela veut dire, réaliser une recherche encore ciblée du modèle. Il existe d'autres propositions d'amélioration de RANSAC comme la version adaptative de [Boulaassal 2010], où le nombre d'itérations est adapté au fur et à mesure de chaque itération, mais notre bonne connaissance *a priori* des données nous permet de

réduire encore plus le nombre d'itérations.

Une fois que la position des rails est déterminée par la méthode décrite, dans l'image suivante nous pouvons filtrer le nuage de points $3D$ pour garder seulement les points appartenant aux rails. Nous proposons de définir deux zones de filtrage en fonction de la position du rail dans l'image précédente. Une représentation de ces deux zones est montrée dans la figure 3.28. Ce filtrage nous permettra de réduire le nombre des points à tester de façon aléatoire et par conséquent augmenter la probabilité W de sélectionner un point appartenant au modèle. Cette approche a été présentée dans un article international et il se trouve dans l'annexe Article ITSC 2012.

Cependant, pour l'évaluation du score des nouveaux modèles, nous gardons tous les points de contours pour déterminer le meilleur modèle en tenant compte de toutes les structures de la voie.

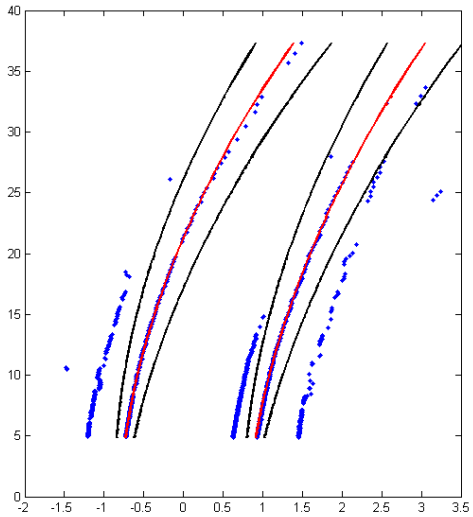


FIGURE 3.28 – Filtrage temporel pour les nuages des points.

Dans cette image nous pouvons voir le filtrage temporel réalisé dans les nuages à partir du modèle de rails détecté dans l'image précéndente. Ce modèle pour chaque rail est représenté en rouge et en noir la zone de filtrage. Tous les points en dehors de cette zone ne seront pas pris en compte pour sélectionner les points aléatoires du processus de RANSAC.

3.3.5 Détermination de la distance maximale de détection

Une fois que les rails ont été extraits, les points de contours retenus peuvent être analysés pour déterminer jusqu'à quelle distance sont visibles les rails avec l'objectif de définir une zone «libre d'obstacles».

Pour pouvoir déterminer la distance maximale, nous avons besoin de connaître la distribution spatiale en $3D$ d'une ligne verticale parfaite dans l'image à partir de la rétro-projection. La figure 3.29 nous montre la courbe que suit la rétro-projection, ou l'écartement entre deux points augmente en fonction de la distance.

Pour pouvoir déterminer la distance maximale de détection, nous utilisons cette résolution. Pour cela, nous avons fixé un seuil. Si dans les points des contours retenus la résolution spatiale d'un point D est 10% supérieure à la résolution de la courbe de la figure 3.29, à partir de cette distance, les autres points sont considérés comme du bruit. Les rails seront seulement extraits jusqu'à la distance D.

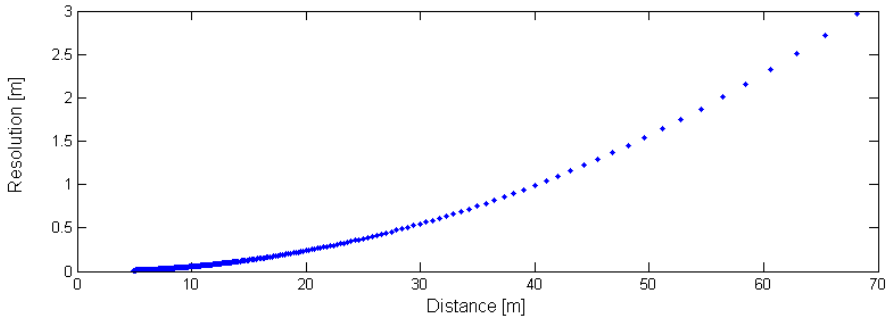


FIGURE 3.29 – Résolution en mètres des pixels verticaux de l'image depuis la partie basse jusqu'à la ligne d'horizon
.

De plus, à partir de cette résolution, nous pouvons analyser les points de contours retenus comme appartenant au modèle *inliers* et déterminer les parties pour lesquelles il existe un manque d'information.

Non seulement cette analyse des points nous permet de déterminer la distance maximale, elle peut se révéler comme un possible détecteur de petits obstacles sur les pistes de roulement. La présence des discontinuités dans les *inliers* peut être occasionnée par des occlusions dues à des obstacles ou simplement à cause des ombres. Considérer les discontinuités des rails comme un possible obstacle sur le rail a été présenté par [Ruder *et al.* 2003] dans la section 2.1 (État de l'art des systèmes de détection d'obstacles ferroviaires).

3.3.5.1 Effets des changement de pente

Un autre limitation de l'utilisation de l'hypothèse du monde plan est l'erreur introduite dans les mesures des distances de la rétro-projection. Le fait que le train se trouve dans un changement de pente introduit des faibles changements dans la position de la projection des rails dans l'image qui peuvent se traduire par des erreurs importants de mesure de distance.

Nous allons analyser les effets introduits dans l'image pour le cas d'un train qui passe d'une pente nulle 0 % à la pente maximale du tracé de la ligne. Dans le chapitre précédent, nous avons extrait des caractéristiques de la voie d'OrlyVal que la pente maximale est de $P = 7\%$ et que le rayon de courbure vertical minimal, celui qui nous permettra d'atteindre la pente maximale plus rapidement est $R_{min} = 600m$.

À partir de la figure 3.30 nous pouvons déterminer la longueur de la corde de raccordement entre l'horizontale et la pente maximale.

Ainsi, de la même figure nous obtenons par trigonométrie, à partir des triangles formés par la prolongation du rayon de courbure et l'intersection avec la pente, que pour n'importe quelle valeur de R, $\psi \approx \alpha$, si nous considérons que $\alpha << 1$, comme montrent les équations de 3.12.

$$\begin{aligned} &triangle\ de\ \psi\ :\ \psi = \forall R = \pi - \beta - \frac{\pi}{2} \\ &triangle\ de\ \beta\ :\ \beta = \pi - \alpha - \frac{\pi}{2} \end{aligned} \tag{3.12}$$

La longueur de l'arc de raccordement s'exprime par l'équation 3.13, en fonction de la pente P et de R :

$$l = \psi R = \mathrm{atan}(P)R \tag{3.13}$$

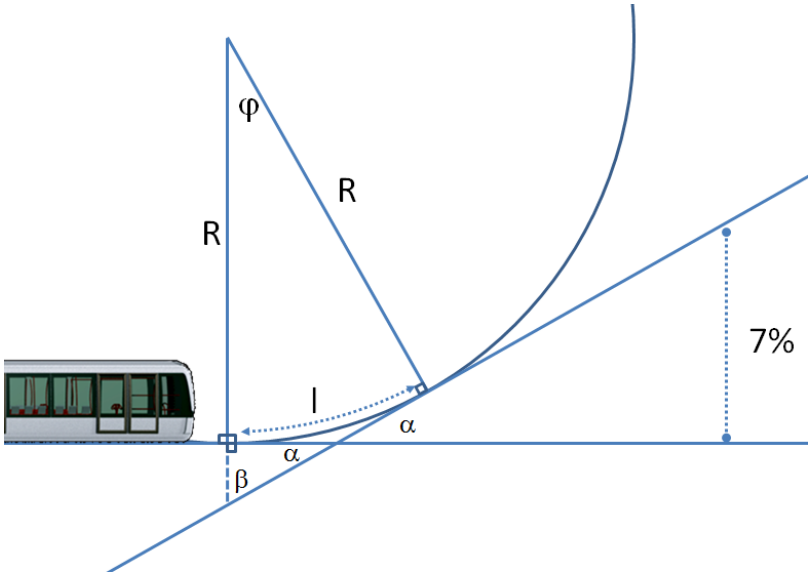


FIGURE 3.30 – Représentation du changement de pente

Une fois connu l'angle de raccordement, nous pouvons représenter les différents cas. La figure 3.31 nous montre d'un côté l'analyse de projection et la rétro-projection de trois cas :

- En bleu, la projection quand la trajectoire est plate, la courbe de sa cote et sa rétro-projection.
- En rouge, le mêmes courbes mais cette fois, il existe une pente positive de 7 % avec un raccordement avec le rayon R_{min} devant le train.
- En vert, pour une pente négative de 7 %, aussi raccordée avec R_{min}.

De l'autre côté, nous avons les rétro-projections des points en considérant l'hypothèse du monde plan. Nous pouvons voir que l'erreur commise en distance, surtout pour le cas d'une pente positive, peut être très important.

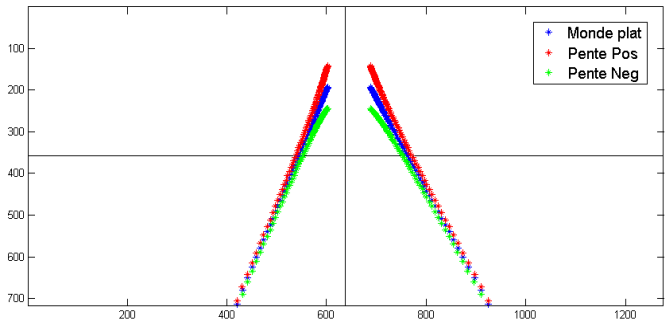
Cependant, puisque dans notre méthode de recherche des points de contour nous appliquons déjà une logique de projection de la largeur des rails, $w_{rail_{pixels}} \pm marge$, dans le cas d'un changement de pente, nous n'allons pas détecter la totalité des rails dans l'image car la largeur n'est pas respectée. Même si cette approche nous limitera en distance de détection, elle nous permettra de réduire l'erreur de détermination de la distance par rétro-projection.

Nous allons analyser la relation entre la marge introduite dans les largeurs des rails dans l'image et les distances de détections.

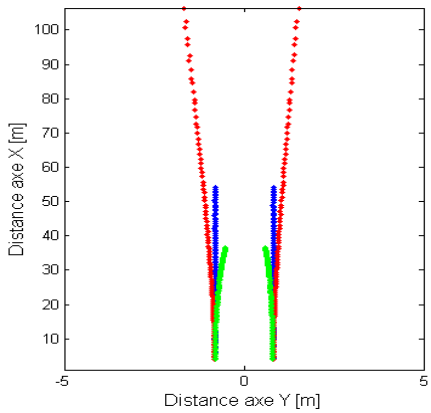
Rappelons la formule 2.13 : $f = \dfrac{T_{image}D}{T_{objet}}$ qui relie la taille de l'objet réel et la taille dans l'image. À partir de cette formule, de la figure 3.31, et en considérant que les rayons de courbure sont supérieurs à la distance de détection, nous pouvons définir les équations 3.14, où nous voyons que la marge en pixels correspond à la marge en mètres dans l'erreur de détection de la distance :

$$\begin{gathered} w_{plat} = \tfrac{k_u f T_{rail}}{D_{plat}}; \quad w_{pente} = \tfrac{k_u f T_{rail}}{D_{pente}}; \quad w_{pente} \leq w_{plat}(1 + Marge); \\ D_{plat} = \tfrac{x}{\sin\theta}; \quad D_{pente} = \tfrac{x'}{sin\theta}; \\ x' = \tfrac{x}{1+Marge} \end{gathered} \tag{3.14}$$

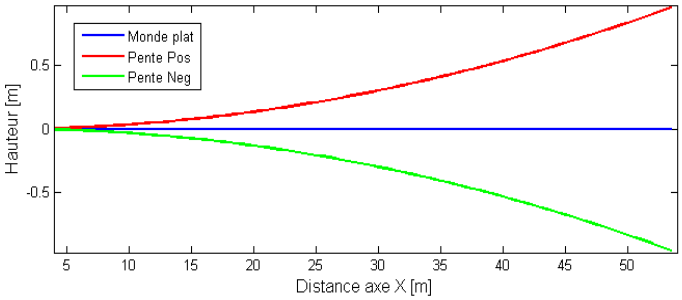
où w est la largeur du rail et k_u est le faceur d'échelle (pixel/mm)



(a) Projection des différentes pentes dans l'image.



(b) Rétro-projection des points en supposant un monde plan.



(c) Cotes des différentes trajectoires projetées.

FIGURE 3.31 – Évaluation de l'erreur de rétro-projection due à l'application de l'hypothèse du monde plan. . Le rayon de raccordement de 1500 mètres et pente finale de 7%.

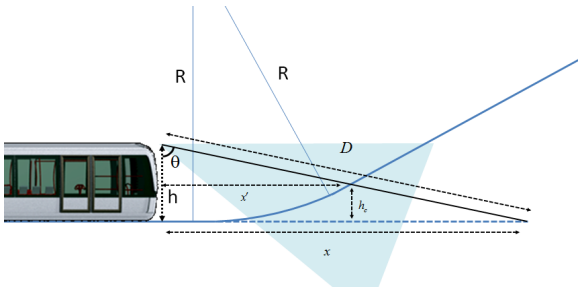


FIGURE 3.32 – Schéma pour trouver la pente maximale supportée.

Par conséquence, la marge utilisée dans la recherche des rails dans l'image est directement la marge d'erreur de détection de la distance définie dans les objectifs du système inférieure à 10 %.

Comme expliqué précédemment, cette contrainte dans la précision de la distance nous limite la distance de détection en rupture de pente. Dans l'équation 3.15 nous pouvons mettre en relation le changement de cote maximum h_c avec le cercle de raccordement de rayon R grâce à la flèche de la corde sous-tendue par l'arc de longueur $2 \times L$, où L correspond à la longueur maximale de détection en pente.

$$\begin{gathered} h_c = \frac{M}{1+M} h; \quad h_c = flèche; \\ L_{max} \approx \frac{corde}{2}; \quad L_{max} = \sqrt{(2 * R - flèche) flèche} \end{gathered} \tag{3.15}$$

Pour nos valeurs de hauteur de la caméra $h \approx 2\ m$, de rayon minimal de courbure verticale $R_{min} = 600\ m$, la longueur maximale de détection en pente est de $L_{max} = 17.77\ m$. La relation obtenue dans l'équation 3.15 nous montre que plus haute est la caméra, plus grande sera la distance de détection en changement de pente pour la même marge d'erreur de détection de distance. Cette distance minimale de détection est supérieur à la distance d'arrêt maximale en arrêt d'urgence de 12 m définie dans le chapitre précédent.

3.4 Résultats

Avec l'objectif de montrer les résultats d'extraction de la voie nous proposons de reconstruire le chemin suivi par le train à partir de l'estimation de la trajectoire de la caméra par rapport à un repère initial. Ce repère peut être par exemple, comme nous l'avons défini, la première image d'une séquence.

Comme le système général doit être autonome, nous ne disposons pas des informations relatives au mouvement du train. Pour cette raison nous avons développé un procédé pour déterminer le mouvement du train à partir des images de la caméra.

Les techniques de localisation visuelle peuvent être divisées en deux groupes : les méthodes basées sur les points d'intérêts et les méthodes directes [Meilland 2012].

Les méthodes basées sur l'extraction de points d'intérêts *feature-based*, cherchent à extraire des primitives visuelles locales dans les images, telles que les points de Harris [Harris & Stephens 1988] ou les SIFT [Lowe 2004]. Une étape d'appariement est ensuite effectuée, c'est-à-dire mettre en correspondance un-à-un les points d'intérêts de chaque image. Pour cela il est nécessaire d'utiliser une mesure de similarité entre les primitives.

Les méthodes directes *image-based* n'ont pas de phase de sélection de points d'intérêt ou de primitives visuelles. Le mouvement de la caméra est directement obtenu en minimisant les erreurs d'intensité communes aux deux images à l'aide d'une transformation paramétrique. Dans ce cas l'estimation du mouvement et la mise en correspondance des pixels s'effectuent simultanément lors de l'optimisation. Dans la majorité des cas, ce type de technique est utilisé pour le suivi d'une surface planaire.

Bien que les méthodes indirectes montrent de très bons résultats, pour le cas ferroviaire il est possible que pour certains cas, il n'existe pas suffisamment des primitives dans les images, par exemple dans les tunnels ou pendant la nuit. Aussi, il faut s'assurer que les primitives détectées sont des points fixes.

Nous avons donc retenu les méthodes *image-based* pour déterminer les déplacements suivis par le train entre chaque image. Puisque ces méthodes sont lourdes en calculs, nous avons défini une zone de recherche réduite plus convenable pour déterminer le déplacement du train. Cette zone est

située immédiatement devant le train et couvre l'entre-axe de la voie. Cette approche montre de très bons résultats dans les travaux ferroviaires de [Chanton 2008]. De plus, dans cette zone la caméra garantit une résolution qui permettra de déterminer le décalage entre deux images successives avec exactitude.

L'approche consiste à utiliser cette portion de l'image et à supprimer la déformation de perspective grâce à l'utilisation de la Transformée en Vue d'Oiseau, car, dans l'image résultante, les lignes parallèles dans le monde réel (les rails) vont rester parallèles.

La zone sélectionnée est définie par la position des rails dans l'image. Une fois déterminée la fonction qui caractérise la courbe de la voie, nous sélectionnons une zone définie entre une distance en mètres de D et $D+1$ avec une largeur suffisante pour couvrir l'entre-axe de la voie. Une image représentative de cette zone et l'image obtenue après la transformée IPM est exposée dans la figure 3.33.

(a) Zone à analyser pour obtenir le déplacement. (b) Image sans déformation perspective utilisée pour le calcul du décalage.

FIGURE 3.33 – Illustration de la méthode proposée pour calculer d'avancement.

De façon que le train avance, dans les images obtenues avec la transformée en vue d'oiseau les rails apparaissent en mouvement. Le décalage entre ces images dépend de l'avancement du train. Un exemple de deux images successives et leur décalage est exposé dans la figure 3.34. Ce décalage peut être déterminé par des méthodes dites *image-based*.

Pour mesurer le déplacement relatif entre deux images successives, nous proposons d'utiliser l'inter-corrélation. Comme les deux images ont été obtenues à partir de la fonction qui définit les rails, le seul déplacement à considérer est vertical, ce qui nous permettra de réduire le coût de calcul de la corrélation.

La fonction d'inter-corrélation, dont est déduit le décalage entre les images est donnée par la formule générale 3.16 :

$$h(p) = \sum_{-\inf}^{\inf} x_i \cdot y_{i-p} \tag{3.16}$$

où x_i et y_i sont les pixels des premier et second vecteurs de mesure. L'algorithme développé a pour objectif de rechercher la valeur du déplacement vertical qui maximise l'opérateur d'inter-corrélation $h(p)$. Pour cela chaque paire de lignes verticales des deux images sera inter-corrélée pour

obtenir le décalage. Un exemple de résultat de $h(p)$ est montré dans la figure 3.35. Dans ce cas, le déplacement déterminé est de 15 pixels.

Connaissant la taille de l'image IPM et sa correspondance en mètres, nous pouvons déterminer l'avancement du train à partir du décalage des images.

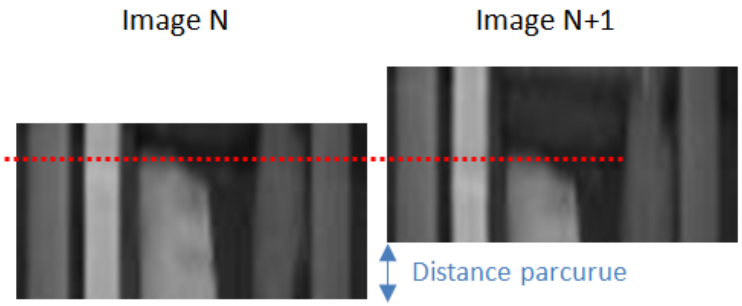


FIGURE 3.34 – Comparaison entre deux images successives pour déterminer l'avancement du train.

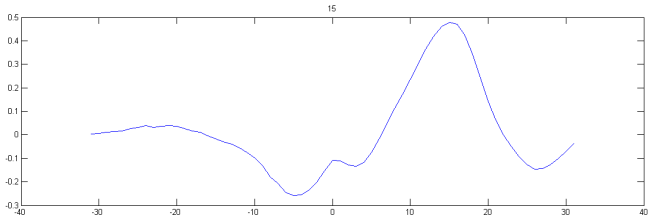


FIGURE 3.35 – Corrélation des lignes verticales des images de la figure 3.34. Le décalage théorique est de 15 pixels.

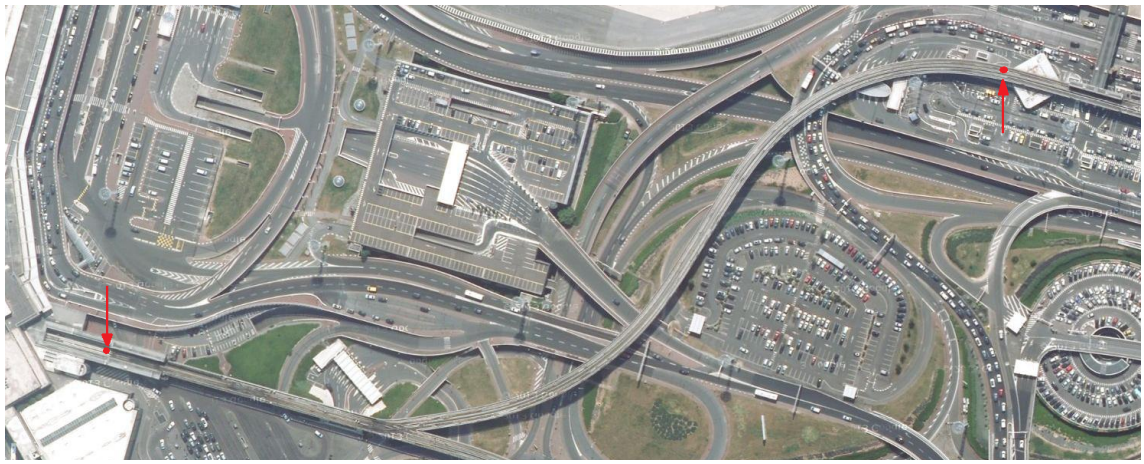


FIGURE 3.36 – Image aérienne du parcours entre Orly Ouest et Orly Sud.
Le début et la fin du parcours sont indiqués par des flèches rouges. Cette partie aérienne du metro OrlyVAL entre les stations OrlyOuest et OrlySud départs et principalement réalisée en viaduc. Le parcours total est de 570 mètres et la vitesse maximale atteinte est de 17 m/s.

La figure 3.38 montre l'extraction des rails sous différentes conditions.

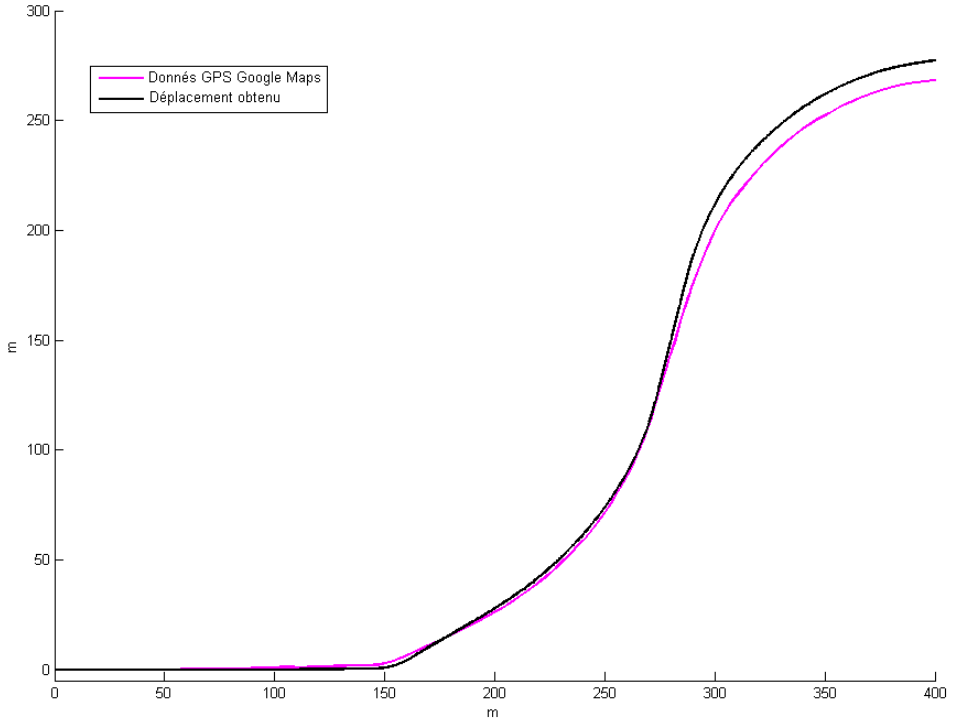
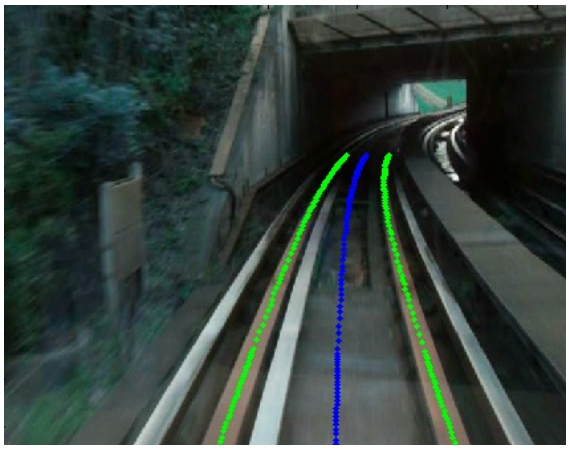


FIGURE 3.37 – Odométrie visuelle obtenue à partir de la détection des rails.
Dans cette image nous pouvons voir le parcours reconstruit par l'odométrie visuelle (en magenta) et le parcours obtenu à partir des données GPS de *Google Maps*. Le décalage en distance finale entre les deux données est de 7.94 mètres pour un parcours total de 570 mètres. Cette erreur peut être due aux approximations de la courbure de la voie par une parabole. Le résultat reste toutefois très semblable au parcours réel du train.

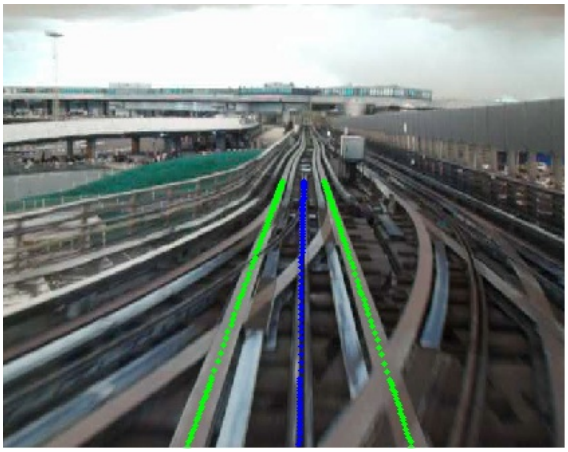
(a) .

(b) .

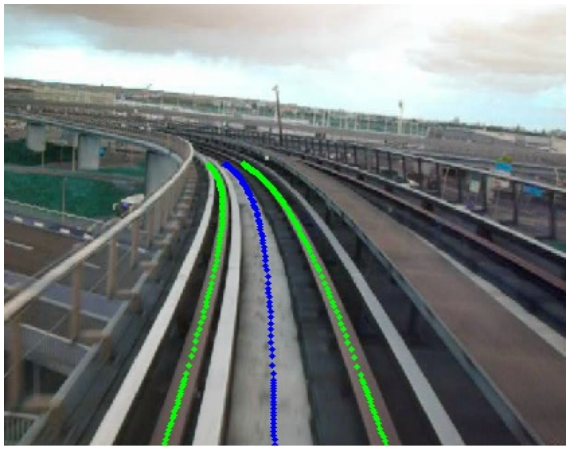
(c) .



(d) .



(e) .



(f) .

FIGURE 3.38 – Résultats de l'extraction des rails pour différents scénarios de voie.

.

Chapitre 4

Détection et caractérisation de croisements

Sommaire

4.1 Motivations **73**
4.2 Les appareils de voie **75**
4.3 Méthode proposée pour la détection des appareils de voie **77**
4.3.1 Utilisation de la transformée *Bird eyes' view* 77
4.3.2 Descripteur : *Template matching* 78
4.3.3 Descripteur : Histogramme de gradients orientés (HOG) 81
4.4 Résultats de détection **84**
4.4.1 Machines à vecteurs de support (SVM) 84
4.4.2 Matrice de confusion et courbe ROC 85
4.4.3 Résultats du descripteur basé en *Template matching* 86
4.4.4 Résultats obtenus avec HOG 87
4.5 Méthode proposée pour la classification du type d'appareil de voie . . . **89**
4.5.1 SVM multi-classe 90
4.5.2 Descripteur proposé : HOG 90
4.5.3 Résultats de classification 92

4.1 Motivations

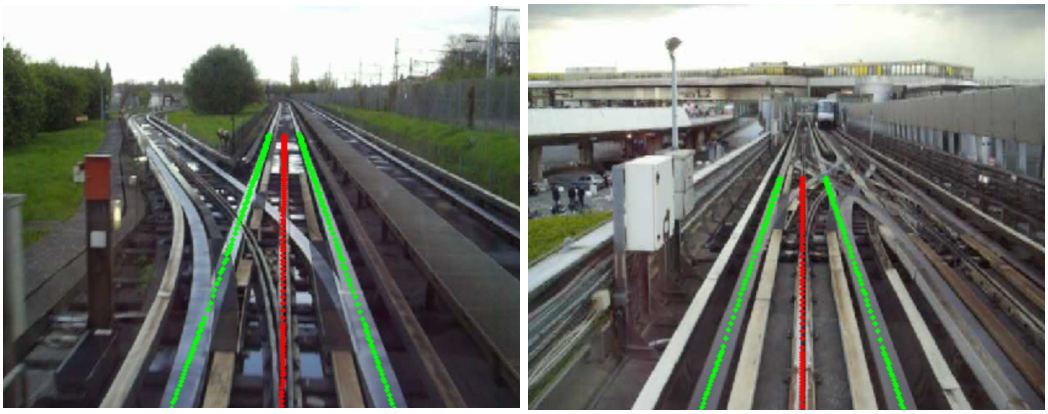
Un point important de notre recherche est la détection des appareils de voie. Elle permettra de déterminer la zone libre d'obstacles sur chacune des 2 voies, tant en aval d'un aiguillage divergent qu'en amont d'un aiguillage convergent (du point de "croisement bon" au coeur d'aiguille). Remarque : contrairement à la voie courante où un "faux négatif" (non reconnaissance des rails) est interprété comme la présence d'un obstacle, la non détection d'un appareil de voie peut "masquer" un obstacle : elle sera donc associée à la détection du feu de signalisation qui protège l'appareil de voie et à la détermination de son état. Ce dernier point n'est pas traité dans ce travail.

D'autre part, pour des raisons pratiques et économiques, pour certains types de lignes ferroviaires il pourrait être envisageable de remplacer une partie ou tout l'équipement de positionnement placé au long de la voie par des capteurs bas coût comme des caméras derrière le pare-brise. La position absolue du véhicule dans une voie peut être déterminée grâce à la détection et la reconnaissance des appareils de voie combinées avec un repérage sur carte numérique, en anglais *map matching*. [Geistler & Bohringer 2004].

Une des limitations de notre système concerne l'arrivée d'un métro à un appareil de voie divergeant. Le système de détection des rails décrit au chapitre précédent est confronté à la difficulté de déterminer deux chemins libres d'obstacles. Pour cela, il devrait détecter les 2 paires de rails. Cependant, la méthode proposée dans nos travaux de thèse pour l'extraction des rails prend en

compte seulement une paire de rails. L'objectif de cette partie de notre étude est de détecter les appareils de voie avant de les traverser. De plus, il faut reconnaitre sa nature, c'est-à-dire, si c'est un appareil de voie convergente ou divergente et sa direction droite ou gauche. Une fois l'appareil de voie détecté et reconnu, nous pouvons modifier la stratégie d'extraction des rails, en cherchant les 2 paires de rails. La figure 4.1 nous montre un exemple de la vue perçue depuis l'avant du métro à l'arrivée de différents appareils de voie.

En outre, le *map matching* pourrait se réaliser à partir de la carte reconstruite par notre système et montrée précédemment dans la figure 3.37 de la section 3.4 et une reconnaissance des appareils de voie. Ainsi, la localisation du métro peut être réalisée par tronçons à partir du système de vision ce qui réduit le nombre des balises nécessaires, ou peut être simplement comme un système de redondance.



(a) Appareil de voie divergent à gauche. (b) Appareil de voie divergent à droite.

FIGURE 4.1 – Ambigüités de détection des rails à l'approche des appareils de voie divergents.

Actuellement, il existe peu de travaux de détection et classification des appareils de voie (en anglais *turnout*). Le plus remarquable est l'étude de [Wohlfeil 2011] réalisée pour la détection des rails et la détection des appareils de voie. Ce système analyse une partie de l'image à la recherche des rails et une fois détectés ils sont appariés. La position des rails dans l'image est transformée en coordonnées globales 3D avec (Z=0), et à partir de ces dernières, il est possible de recréer une carte des rails et reconnaitre la présence des appareils de voie. La figure 4.2 montre des exemples de cette technique. L'une des principales limitations de cette approche est que l'analyse s'effectue sur une partie réduite de l'image, car la méthode de détection des gradients est lourde en calcul puisqu'elle est basée sur les valeurs propres de l'image en niveau de gris. De plus, les appareils de voie sont détectés à partir de la carte reconstruite, et pas directement dans l'image.

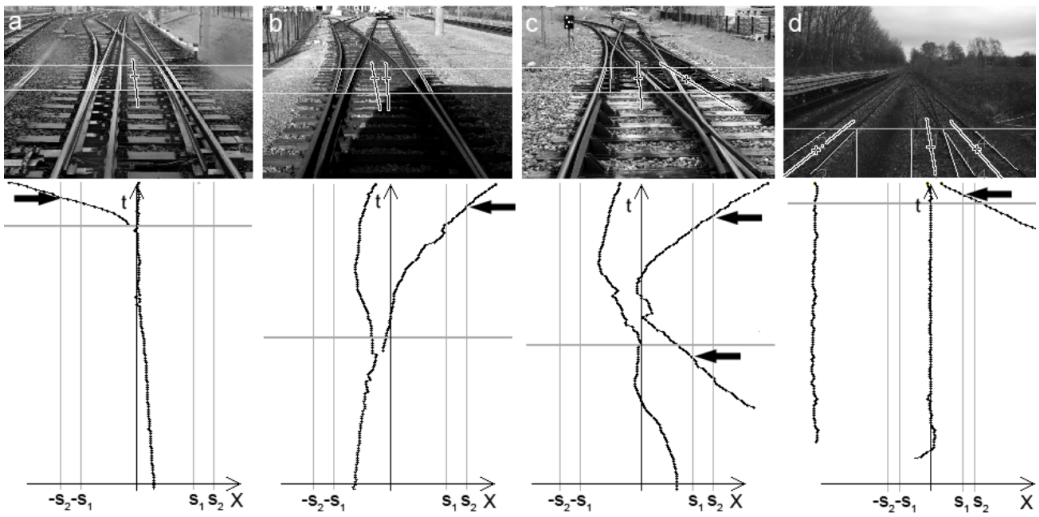


FIGURE 4.2 – Détection des aiguilles à partir de la carte 3D (Z=0) reconstruite. Image extraite de [Wohlfeil 2011].

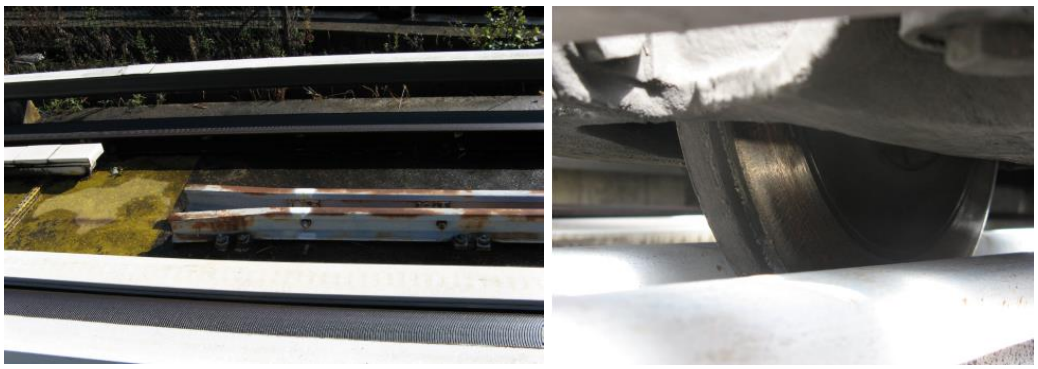
4.2 Les appareils de voie

Les appareils de voie permettent d'aiguiller un véhicule sur une autre voie. Pour les métros VAL, les appareils de voie sont de type guidage central comme le montre la figure 4.3.

Le système à guidage central consiste à disposer dans l'axe de la voie une ornière située sous le niveau des pistes de roulement, dans laquelle s'engage un galet, à axe horizontal, solidaire du cadre de l'essieu du véhicule.

L'ornière est constituée de deux rails fixés sur le radier et espacés au niveau des champignons de 5 cm, d'une aiguille en acier de 6 mètres de long et d'organes de commandes (un groupe de manœuvre hydraulique et une armoire de commande).

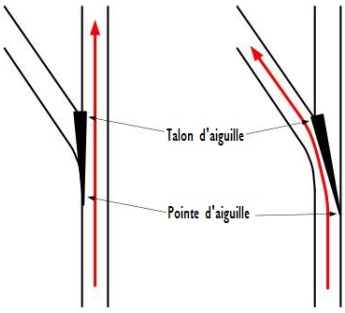
Les entrées et les sorties de l'ornière s'effectuent par des sifflets de 20 *cm* d'ouverture qui tiennent compte des cas les plus défavorables de présentation des galets dans l'ornière. La partie supérieure de l'ornière est au même niveau que les pistes de roulement.



(a) Exemple d'ornière ou rail central de guidage. (b) Ornière avec un galet engagé.

FIGURE 4.3 – Représentation des éléments d'un appareil de voie.

Le choix de la direction s'effectue par une aiguille située à l'intérieur de l'ornière et qui oriente le



(a) Principe de fonctionnement de l'aiguille insérée dans l'ornière

(b) Vue en détail d'une aiguille.

FIGURE 4.4 – Représentation d'un aiguille et son schéma.

véhicule en voie directe ou en voie déviée. Le schéma de la figure 4.4 illustre ce principe. Nous pouvons nous rendre compte que déterminer le positionnement de l'aiguille depuis la cabine du conducteur n'est pas une tâche facile. En conséquence, à l'arrivée d'un appareil de voie divergent, le système de vision ne peut pas déterminer vers quelle direction est disposée l'aiguille avant de traverser l'appareil de voie. Ceci est une des raisons pour lesquelles nous avons développé un système de détection et de classification d'aiguilles.

Les différents types d'appareils de voie présents sur la ligne OrlyVAL sont :
- les branchements, qui comportent un seul appareil de voie ;
- les communications de voie (simples ou croisées), qui comportent 2 appareils de voie ;
- la jonction en Y.

Cependant,la méthode proposée a été pensée pour pouvoir être adaptable aux voies ferrées classiques, par exemple en reconnaissant les 2 rails côte-à-côte que l'on voit bien sur la figure 4.2.

Le rail de guidage en station

Un cas particulier d'usage d'un rail central hors les appareils de voie c'est «le rail central en station». Ce rail a pour objectif de limiter la rotation de la caisse dans les zones de station pour permettre un ajustement du gabarit des véhicules par rapport aux quais, en particulier en cas de crevaison d'un pneu porteur ou de guidage. La figure 4.5 montre un exemple de ce rail.

FIGURE 4.5 – Rail de guidage en station

4.3 Méthode proposée pour la détection des appareils de voie

Puisque dans le cas d'étude, la voie à analyser se caractérise par la présence d'un rail central de guidage dans les appareils de voie qui permet de faire basculer le métro d'une piste de roulement à l'autre, cette caractéristique nous permettra déterminer à partir du système de vision que le métro s'approche d'un appareil de voie.

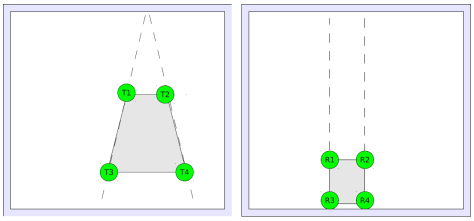
Pour faciliter la tâche de détection de ce rail dans l'image, nous proposons de tirer profit de la détection des rails. Ceci nous permettra de réduire la zone de recherche du rail central de guidage. Ainsi, une fois déterminée la position des rails, nous pouvons analyser la morphologie de l'entrevoie à une certaine distance à l'avant du métro. L'objectif est d'identifier la présence du rail de guidage central avant d'y arriver. Cette distance sera choisie en fonction de la résolution spatiale de la caméra.

Puisque la morphologie de la voie est essentiellement constante, l'utilisation d'un descripteur local peut donner de très bons résultats. L'analyse des descripteurs locaux basés sur des histogrammes de gradients orientés en anglais *Histogram of Oriented Gradients*(HOG) est très répandue et efficace comme le montrent les expériences obtenues au sein du CAOR [Zaklouta & Stanciulescu 2011] et [Zaklouta *et al.* 2011] réalisées pour la caractérisation et classification des panneaux routiers. En conséquence, nous proposons d'utiliser les HOG et le *template matching* comme techniques de caractérisation de la région d'intérêt. Ces deux descripteurs sont actuellement utilisés couramment par la communauté de reconnaissance de formes pour la détection ou la reconnaissance d'objets.

4.3.1 Utilisation de la transformée *Bird eyes' view*

Avec l'objectif d'homogénéiser les images des différents types d'aiguilles, nous proposons d'obtenir une image en vue vol d'oiseau d'une portion de la voie à l'avant du métro. Cette portion rectangulaire aura une largeur qui couvrira un peu plus que l'écartement des rails et 1 mètre de hauteur. À partir de la transformée en vue d'oiseau, nous pouvons obtenir des images IPM (*Inverse Perspective Mapping*) dans lesquelles la déformation en perspective a disparu, comme le montre la figure 4.6. L'effet d'avoir des images de la même taille et avec des caractéristiques similaires nous

permettra de mieux différencier la morphologie de l'entrevoie pour les images qui contiennent le rail central de guidage et les images sans.



(a) Image en perspective. (b) Perspective supprimée.

FIGURE 4.6 – Transformation en vue d'oiseau. Image extraite de [Maire & Bigdeli 2010]

Un exemple de la zone choisie pour déterminer la présence d'un appareil de voie est présenté dans la figure 4.7 par un rectangle rouge. Les images IPM obtenues sont aussi représentées et nous pouvons voir comment, à partir de cette transformation les images deviennent homogènes et indépendantes de leur forme d'origine (ligne droite, virage, etc).

4.3.2 Descripteur : *Template matching*

L'approche par corrélation (*template matching*, en anglais)[Brunelli 2009] est basée sur une mesure de similarité entre l'image et un modèle de l'objet recherché. Cette technique a été couramment utilisée pour la détection d'obstacles par vision, reconnaissance faciale et même pour des tâches de comptage d'objets dans le milieu industriel [Dharpure *et al.* 2013] et [Aksoy *et al.* 2004].

Cette technique cherche à mettre en évidence les régions de l'image ressemblant le plus à l'objet recherché par le calcul d'une mesure de corrélation. Son principe est très simple : il consiste en la mise en correspondance d'un modèle de l'objet défini par un *template* T, qui n'est rien d'autre qu'une image de l'objet lui-même, par convolution du modèle avec l'image I.

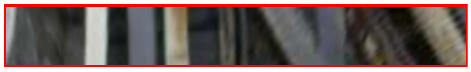
Pour chercher le *template* $T(x, y)$ de taille (n_x, n_y) tout au long des deux dimensions de l'image $I(x, y)$ de taille m_x, m_y où $n_x < m_x$ et $n_y < m_y$, la corrélation croisée normalisée $\gamma(u, v)$ est calculée pour chaque point (u, v) pour I et T comme le montre l'équation 4.1 :

$$\gamma(u,v) = \frac{\sum_x \sum_y [I(x,y) - \mu(I(u,v)).[T(x-u, y-v) - \mu(T)]]}{\sqrt{\sum_x \sum_y [I[(x,y) - \mu(I(u,v))]^2 . \sum_x \sum_y [T(x-u, y-v) - \mu(T)]^2}} \tag{4.1}$$

où $u \in 1, ..., m_x - n_x + 1$, $v \in 1, ..., m_y - n_y + 1$ et $\mu(I(u, v))$ représente la valeur moyenne de $I(x, y)$ dans la section à corréler, et $\mu(T)$ la moyenne du *template* T. Puisque le dénominateur de 4.1 est la variance des deux images $I(x, y) - \mu(I(u, v)$ et $T(x-u, y-u) - \mu(T)$, la corrélation croisée normalisée est particulièrement efficace pour comparer l'image et le modèle, car elle est robuste aux variations locales d'intensité.

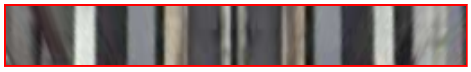
L'intervalle des résultats de $\gamma(u, v)$ est $[-1.0, 1.0]$, où une valeur de 1 correspond à une parfaite correspondance. Ceci fait que pour déterminer la présence d'un objet il faut définir un seuil de ressemblance. Un exemple de corrélation croisée bidimensionnelle est montré dans la figure 4.8. Le maximum de la figure 4.8c représente la position (u, v) pour laquelle la correspondance entre l'image I et le *template* T est maximale.

(a) Image en perspective avec un virage à gauche.



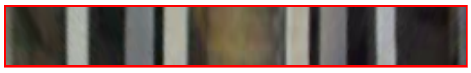
(b) IPM reconstruit.

(c) Image en perspective avec un virage à droite



(d) IPM reconstruit.

(e) Image en perspective en ligne droite.

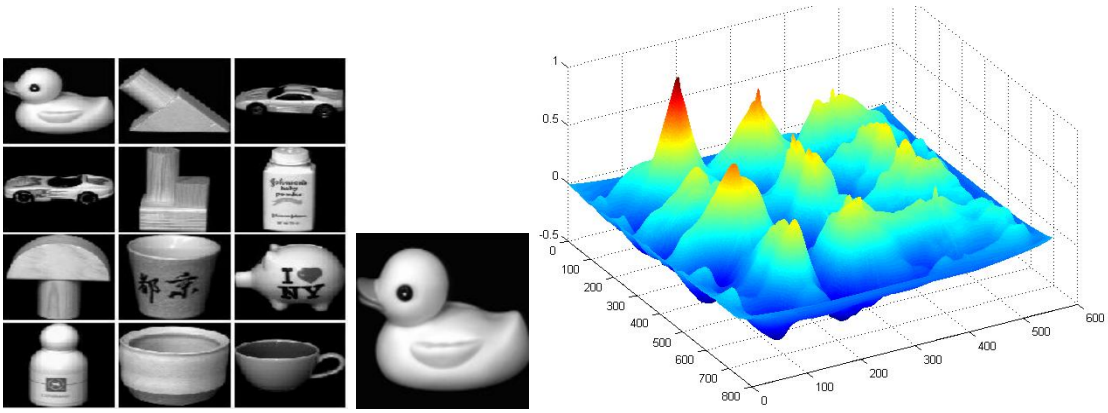


(f) IPM reconstruit.

FIGURE 4.7 – Extraction de la zone de recherche des appareils de voie et l'image IPM obtenue.

Cependant, le (*template matching* souffre d'une sensibilité à la variance intra-classe et aux changements de points de vue. De plus, ces méthodes relativement simples à mettre en œuvre sont lourdes en temps de calcul.

Pour surmonter ces inconvénients, nous avons profité de la connaissance *a-priori* de la voie et de l'homogénéisation obtenue avec les images IPM. C'est-à-dire, les déplacements (u, v) suivis par le

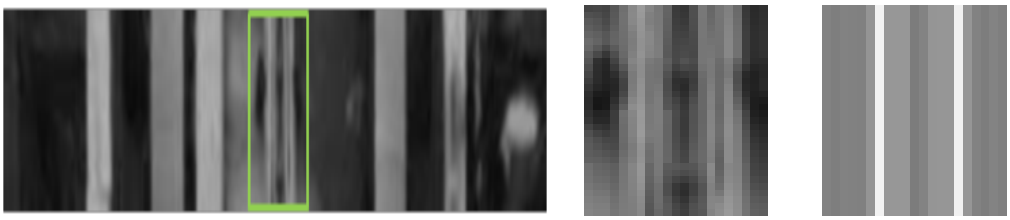


(a) Image I (b) *Template* (c) Plan de corrélation croisée $\gamma(u, v)$

FIGURE 4.8 – Résultat de la corrélation croisée normalisée.

template vont être réduits au minimum, et puisque tous les appareils de voie sont équipés du rail de guidage central ou ornière, c'est seulement la partie centrale de ces images qui sera analysée. Nous avons défini comme objet le rail central et nous l'avons associé à un *template* T, et les déplacements (u, v) se limiteront aux alentours de la position théorique du rail de guidage dans la partie centrale de l'image IMP. De plus, puisque toutes les images auront la même taille et que le rail central occupera tout ou une partie de l'image, les déplacements verticaux de T ne seront plus nécessaires, ce qui réduit considérablement le temps de calcul.

La figure 4.9 montre un exemple d'image IPM, de la partie d'image à analyser et du *template* T.



(a) section du vue d'oisseau (b) Partie centrale (c) Template

FIGURE 4.9 – Transformation en vue d'oiseau

Comme indiqué précédemment, plus la valeur de $\gamma(u, v)$ est proche de 1, plus la ressemblance entre l'image et le modèle est grande. En profitant des caractéristiques de l'objet à reconnaitre, à savoir deux rails verticaux écartés d'une petite distance, nous pouvons voir sur la figure **??** que l'auto-corrélation du *template* a une forme spécifique. Les deux rails de l'ornière font apparaitre deux pics symétriques de chaque côté du pic principal comme on peut le voir sur la figure 4.10. Le pic principal correspond à la position (u, v) pour laquelle la ressemblance est parfaite, et les deux pics latéraux, correspondent aux déplacements où le rail opposé du *template* déplacé est superposé au rail opposé de l'image.

Donc, à partir de la réponse d'auto-corrélation nous proposons d'utiliser comme descripteur la somme de ces 3 sommets moins la somme des valeurs de 2 vallées, comme indiqué dans l'équation 4.2. Puisqu'il faut déterminer un seuil de ressemblance, notre objectif avec ce descripteur est de profiter de la forme de notre objet pour améliorer les résultats de détection.

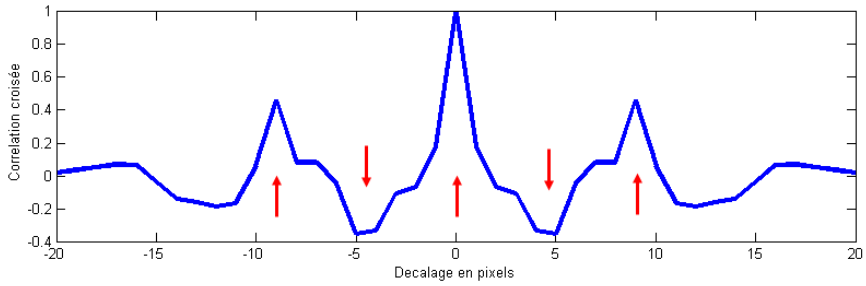


FIGURE 4.10 – Auto-corrélation du *template*.

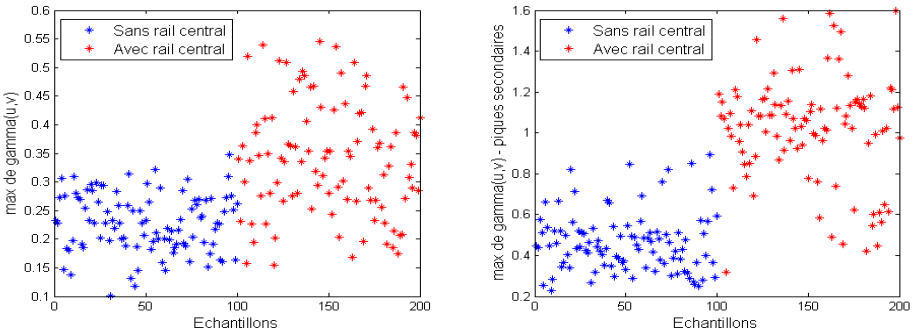


FIGURE 4.11 – Comparative entre la valeur maximale de la corrélation croisée et le descripteur proposé pour une série de corrélations croisées. Les points bleus représentent le résultat pour des images sans rail de guidage, et les points rouges, les images dans lesquelles il y en a.

$$Descripteur = \sum_{i=1}^{3} pic - \sum_{i=1}^{2} vallée \tag{4.2}$$

Comme attendu, les figures de 4.11 montrent l'amélioration du pouvoir de discrimination entre la simple corrélation croisée et le descripteur proposé.

4.3.3 Descripteur : Histogramme de gradients orientés (HOG)

Ce descripteur proposé par [Dalal & Triggs 2005] est utilisé principalement en vision par ordinateur pour la détection de piétons et d'objets. Il s'est révélé particulièrement efficace pour la détection de personnes, car l'apparence locale et la forme d'un objet peuvent être décrites par la distribution d'intensité et l'orientation de ses gradients.

L'histogramme est un outil qui nous permet de représenter la répartition d'une variable sur une population. Dans le cas de HOG, cette variable est l'orientation des gradients et la population l'ensemble des pixels de l'image. HOG définit alors dans une région (bloc) les proportions de pixels dont l'orientation du gradient appartient à un certain intervalle (*bin*). Ces proportions caractérisent la forme présente dans cette région au point que le HOG peut alors être utilisé en tant que descripteur de forme.

Pour obtenir ce descripteur, l'image est divisée de façon dense en $N_{colonne} \times M_{ligne}$ blocs, et ces blocs sont ensuite divisés à la fois en cellules. Le nombre de blocs et de cellules sont alors des variables à choisir dès la première étape de l'algorithme. Un exemple d'une division est montré dans l'image 4.12a.

(a) Division de l'image en blocs et cellules.

(b) Résultat de détection des personnes à partir de HOG

FIGURE 4.12 – Exemple de l'extraction de HOG pour la détection des piétons.

Pour chaque pixel de l'image nous appliquons un masque H(Sobel) puis H^T pour obtenir les gradients horizontaux et verticaux G_x et G_y respectivement.

$$G_x = \begin{bmatrix} +1 & 0 & -1 \\ +2 & 0 & -2 \\ +1 & 0 & -1 \end{bmatrix} * I \quad et \quad G_y = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} * I \tag{4.3}$$

On obtient la magnitude du gradient $G = \sqrt{{G_x}^2 + {G_y}^2}$ l'orientation des gradients $\Theta = \tan^{-1}(G_x, G_y)$. Le vecteur des caractéristiques est ensuite formé pour chaque cellule, c'est-à-dire, des histogrammes des gradients orientés dans la cellule où l'amplitude sert de poids. Puis on concatène les histogrammes des cellules dans le bloc pour finalement, normalisé en intensité chaque bloc.

Un bon compromis doit être trouvé entre la taille des cellules et la taille des blocs. En effet, si le nombre de cellules dans un bloc est élevé, le descripteur est précis. Aussi, pour obtenir plus d'information, il est recommandé de chevaucher les blocs [Dalal & Triggs 2005].

Comme la taille de l'image IPM est déjà réduite, nous avons décidé d'utiliser une seule cellule par bloc pour obtenir le descripteur de HOG souhaité. Nous avons testé les combinaisons du nombre de colonnes/lignes pour déterminer la meilleure combinaison pour la détection.

Quant à l'extraction des gradients de l'image, au lieu d'utiliser Sobel, nous avons choisi d'utiliser l'opérateur développé par [Ando 2000] car il a été reconnu pour être optimum à l'extraction de la norme et l'orientation des gradients. L'équation 4.4 montre le masque de cet opérateur 3×3.

$$M = \begin{bmatrix} -0.112737 & 0 & 0.112737 \\ -.274526 & 0 & .274526 \\ -0.112737 & 0 & 0.112737 \end{bmatrix} \tag{4.4}$$

Puisque notre motif présente un axe de symétrie, nous pouvons limiter les orientations des gradients à l'intervalle $[0..\pi]$. Cet espace a été ensuite divisé en $N_{bin} = 5$ intervalles, et plus précisément, puisque nous cherchons à déterminer la présence du rail de guidage, nous avons choisi les intervalles suivantes $[0-15; 15-80; 80-100; 100-165; 165-180]^\circ$ comme montré dans la figure 4.13.

Les figures 4.14 et 4.15 nous montrent l'extraction du descripteur HOG pour la partie centrale des images IMP. La figure 4.14b représente HOG grâce à un graphe des flèches pour chaque bloc, où la direction correspond à l'orientation de l'intervalle et l'amplitude au résultat de la normalisation du bloc. Puis 4.14c montre le même descripteur dans sa forme d'histogrammes. Dans cet exemple, nous montrons le descripteur de HOG pour Extraction de HOG 3×1 et une seule cellule par bloc.

Nous pouvons voir que l'amplitude de l'intervalle $[80^\circ - 100^\circ]$ est faible quand les images ne contiennent pas le rail central, alors que pour les images IPM des appareils de voie, l'amplitude de cet intervalle est plus importante que les autres.

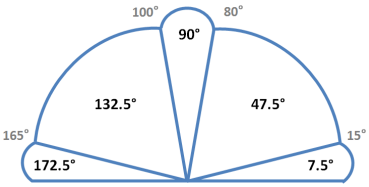
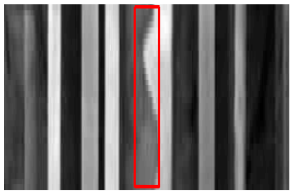
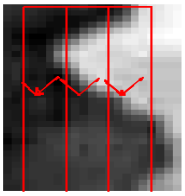


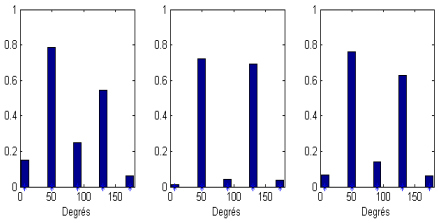
FIGURE 4.13 – Intervalles définis pour former l'histogramme de HOG



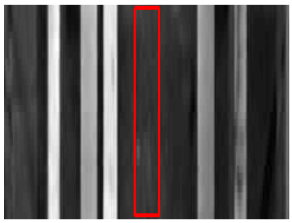
(a) IPM de la zone à analyser



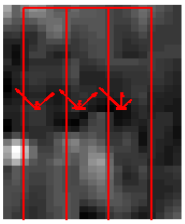
(b) HOG avec flèches.



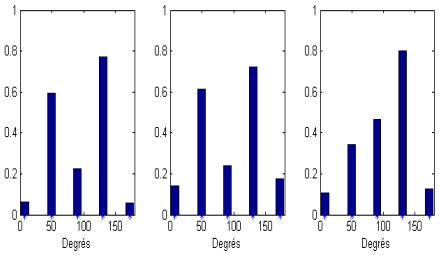
(c) Histogrammes de HOG.



(d) IPM de la zone à analyser

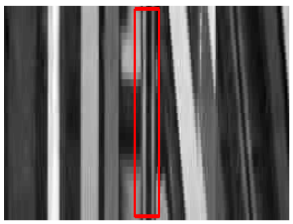


(e) HOG avec flèches

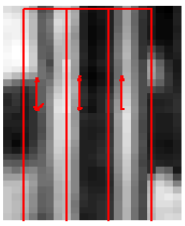


(f) Histogrammes de HOG

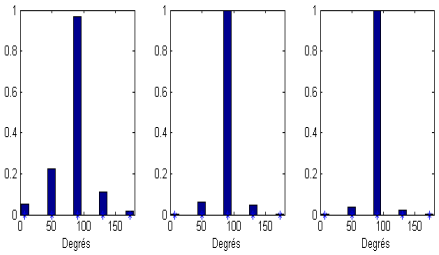
FIGURE 4.14 – Extraction de HOG 3 colonnes pour des images IMP sans rail de guidage.



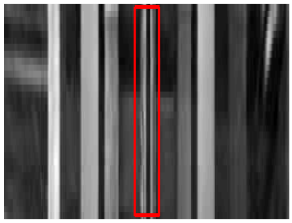
(a) IPM de la zone à analyser



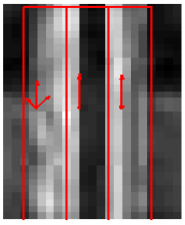
(b) HOG avec flèches



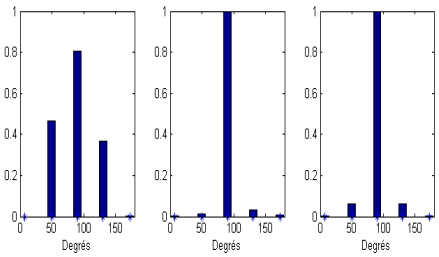
(c) Histogrammes de HOG



(d) IPM de la zone à analyser



(e) HOG avec flèches



(f) Histogrammes de HOG

FIGURE 4.15 – Extraction de HOG 3 colonnes pour des images IMP avec rail de guidage.

4.4 Résultats de détection

Nous avons évalué nos descripteurs avec la base d'images obtenue lors de nos enregistrements dans l'OrlyVAL et à Rennes. Cette base de données contient des exemples de différents appareils de voie aux apparences différentes et dans des scènes et des conditions de luminosité très diverses (voir figure 4.28).

Le nombre total des images IMP est de 9515 dont 4949 sont des images extraites dans des appareils de voie. Pour l'apprentissage supervisé nous avons utilisé seulement 30% de la base de données.

Pour évaluer nos descripteurs, nous allons les tester séparément et ensemble en utilisant le classificateur «Machines à vecteur de support» avec différents types de fonctions de *kernel*. Les résultats seront analysés grâce à la matrice de confusion et les courbes ROC.

4.4.1 Machines à vecteurs de support (SVM)

Les machines à vecteurs de support ou séparateurs à vaste marge (en anglais *Supports Vectors Machines* SVM), sont des classificateurs reposant sur un processus d'apprentissage développé dans les années 1990 et sont largement utilisées depuis pour la reconnaissance de formes, la régression et l'estimation de densité. Le principe des SVM consiste à déterminer une fonction limite, séparant au mieux les deux classes. Pour optimiser la séparation, SVM cherche l'hyperplan pour lequel la distance entre la frontière des deux groupes et les points les plus proches est maximale, c'est le principe de maximisation de la marge.

Nous allons développer brièvement ces différents points dans les paragraphes qui suivent.

Pour la classification de données, considérons l'espace R_N des échantillons. Un point dans cet espace sera noté $x = (x_1; x_2; ...; x_N)$. Dans le cas qui nous intéresse, la détection du rail central, la classification est binaire : la classe objet (rail de guidage) et la classe non-objet (non-rail ou entre-voie). Pour des données séparables, l'équation f de l'hyperplan qui sépare les classes peut être écrite par 4.5 et l'hyperplan est calculé de façon à maximiser sa marge avec les données d'apprentissage positives et négatives.

$$f(x) = w^T x + b \tag{4.5}$$

où w est un coefficient vectoriel et b est un coefficient scalaire. Cette fonction définit, via le vecteur w, un hyperplan dans l'espace R_N. Ce type de classificateur est dit linéaire. Plusieurs hyperplans conviennent, parmi eux, il en existe un unique qui permet d'obtenir une marge de séparation maximale entre les deux classes de points. La marge est la distance minimale entre la surface de décision et les échantillons x_i (qui sont supposés être tous correctement classifiés par cette surface de décision). La classe de ces échantillons est notée y_i, avec y_i = 1 s'il s'agit d'un objet et y_i = -1 sinon.

Par ailleurs, SVM autorise les techniques dites *kernel-methods*. Ces techniques utilisées dans le cas de données non-séparables (c'est-à-dire au cas où les échantillons donnés durant l'apprentissage ne peuvent être séparés par un hyperplan), permettent de projeter les données dans des espaces de dimensions supérieures, de façon implicite via une fonction définie par un noyau ou *kernel* ϕ.

L'implémentation SVM du classificateur utilisé dans nos travaux est basée sur la librairie LIBSVM [Chang & Lin 2011].

4.4.2 Matrice de confusion et courbe ROC

La matrice de confusion est un tableau à double entrée calculé en comparant les résultats issus de la classification et les classes d'apprentissage définies.

Sur la diagonale, on trouve donc les valeurs bien classées et hors de la diagonale les éléments mal classés. Les colonnes de la matrice représentent les classes d'apprentissage (vérité terrain), les lignes, les classes issues de la classification. La somme des valeurs sur une colonne donne le nombre d'exemplaires théoriques de la catégorie. Un exemple de matrice de confusion est montré dans le tableau 4.1.

Tableau 4.1 Matrice de confusion théorique

		Condition		
		Positive	**Négative**	
Test	**Positif**	**Vrai Positif (VP)**	**Faux Positif (FP)**	Précision
	Négatif	**Faux Négatif (FN)**	**Vrai Négatif (VN)**	Valeur Prédictive Négative
		Sensibilité	Spécificité	Exactitude

Du tableau 4.1, nous détaillons chacune des variables statistiques nécessaires pour évaluer nos résultats :

$$Sensibilité = Rappel = \frac{VP}{VP + FN} \tag{4.6}$$

$$Spécificité = \frac{VN}{VN + FP} \tag{4.7}$$

$$Précision = \frac{VP}{VP + FP} \tag{4.8}$$

$$Valeur\ Predictive\ Négative = \frac{VN}{VN + FN} \tag{4.9}$$

$$Exactitude = \frac{VP + VN}{VP + FP + VN + FN} \tag{4.10}$$

La *sensibilité* d'un test mesure sa capacité à donner un résultat positif lorsqu'une hypothèse est vérifiée. Elle s'oppose à la *spécificité*, qui mesure la capacité d'un test à donner un résultat négatif lorsque l'hypothèse n'est pas vérifiée. La valeur prédictive d'un test est la probabilité qu'une condition soit présente en fonction du résultat de ce test. La *valeur prédictive positive* ou *précision* est la probabilité que la condition soit présente lorsque le test est positif. La *valeur prédictive négative* est la probabilité que la condition ne soit pas présente lorsque le test est négatif. Finalement *l'exactitude* globale du système représente la capacité de bien classifier les classes cependant, peut être une mesure ambigüe si les classes ne sont pas pondérées.

La courbe ROC de l'anglais *Receiver Operating Characteristic* [Delacour *et al.* 2005] est un outil d'évaluation et comparaison des modèles de classification indépendants qui permet de visualiser les performances d'un seul coup d'œil et opérationnel même dans les cas des distributions très déséquilibrés.

L'idée de la courbe ROC est de faire varier le «seuil» de 1 à 0. Pour chacune de ces variations, on calcule la sensibilité et la spécificité de l'examen au seuil fixé. Chaque valeur du seuil est associée à un couple (Sensibilité, Spécificité) que l'on reporte dans un graphique sur un plan orthonormé où les abscisses (x) représentent (1- Spécificité) variant de 0 à 1 et où les ordonnées (y) représentent (Sensibilité). Ce graphe théorique est montré dans la figure 4.16.

À la vue de la courbe théorique, un test avec un fort pouvoir discriminant occupera la partie supérieure gauche du graphique. Un test avec un pouvoir discriminant moins puissant montrera une courbe ROC qui s'aplatira vers la première diagonale du graphique.

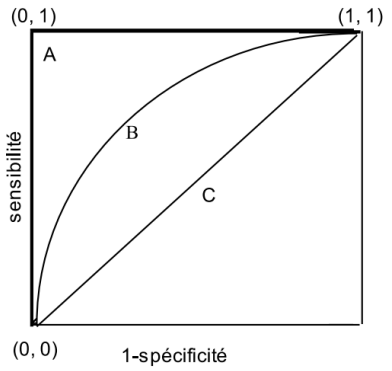


FIGURE 4.16 – Trois scénarios possibles d'une courbe ROC
Un test idéal (A) donne une séparation nette entre les deux classes offrant ainsi une courbe ROC passant par le coin supérieur gauche. Un test sans aucune discrimination (C) donne une courbe allant du point de coordonnées (0,0) au point (1,1). Habituellement, la courbe passe quelque part entre ces deux extrêmes (B).

Nous utiliserons donc la courbe pour décider quel est le seuil qui offre le meilleur compromis entre sensibilité et spécificité. Il s'agira du seuil où la courbe ROC montre un point d'inflexion.

Grâce à l'analyse des courbes ROC des différents classifieurs, nous pourrons pour mettre en évidence quel est le test qui possède le meilleur pouvoir discriminant.

4.4.3 Résultats du descripteur basé en *Template matching*

Pour déterminer la capacité de détection du descripteur proposé, nous allons le comparer avec une corrélation croisée simple. C'est-à-dire, notre descripteur $D = \sum_{i=1}^{3} pic - \sum_{i=1}^{2} vallée$ en comparaison avec la valeur du maximale de $\gamma(u, v)$ (*Template Matching classique*).

Le type de classifieur utilisé est un SVM linéaire. La base de données a été étiquetée manuellement, et pour l'entraînement nous avons choisi d'utiliser seulement 30 % des images. Cela nous fait un total de 1369 images avec rail central de guidage et 1484 sans rail de guidage pour la phase d'apprentissage et 3464 et 3196 pour la phase de test respectivement. Il était possible d'ajouter d'autres images sans rail de guidage, mais cela n'a pas été fait pour équilibrer les échantillons.

La figure 4.17 illustre les courbes ROC des deux descripteurs. Nous pouvons voir que le descripteur proposé présente des résultats de classification meilleurs et dépasse le pouvoir discriminant du simple *Template matching*.

La matrice de confusion pour le résultat de la simple corrélation croisée est montrée dans le tableau 4.2 et la matrice de confusion pour notre descripteur dans la table 4.3.

Tableau 4.2 Matrice de confusion pour la corrélation croisée

		Condition		
		Positive	Négative	
Test	Positif	2770	477	0.8531
	Négatif	694	2719	0.7967
		0.7997	0.8508	0.8242

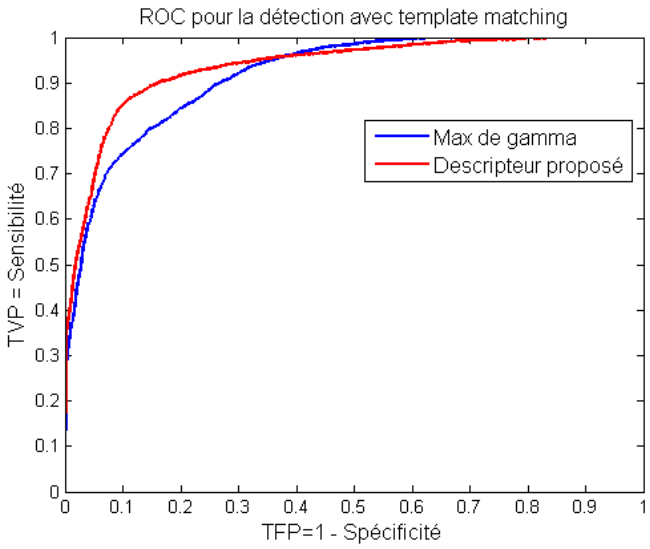


FIGURE 4.17 – Courbe ROC comparative entre le *Template Matching* et le descripteur proposé
La courbe bleue répresente la ROC pour l'utilisation du *Template Matching* classique, et la courbe rouge représente la ROC pour le descripteur proposé

Tableau 4.3 Matrice de confusion pour le descripteur proposé

		Condition		
		Positive	Négative	
Test	Positif	3002	358	0.8935
	Négatif	462	2838	0.86
		0.8666	0.888	0.8769

4.4.4 Résultats obtenus avec HOG

Pour le descripteur HOG, nous avons réalisé différents tests pour pouvoir déterminer le meilleur nombre des blocs de HOG. Vu que nous cherchons à caractériser le rail central de guidage, le nombre des lignes et colonnes de HOG a varié entre 1 et 3. La figure 4.18 montre un *zoom* de la courbe ROC pour les différentes combinaisons de lignes et colonnes. Les trois compositions des blocs *(colonne* $\times$ *ligne)* qui donnent les meilleurs résultats sont 3×1, 3×2 et 3×3.

Étant donné que la taille du descripteur augmente avec le nombre de blocs, et que les résultats obtenus sont semblables pour différents descripteurs, on retiendra le descripteur le plus simple.

Les résultats de détection pour les descripteurs HOG 3×1 et 3×2 sont montrés dans les matrices de confusion 4.4 et 4.5 respectivement. Le seuil de classification utilisé est celui qui porte le meilleur compromis entre sensibilité et spécificité déterminé à partir des courbes ROC.

Tableau 4.4 Matrice de confusion pour le descripteur HOG 3×1

		Condition		
		Positive	Négative	
Test	Positif	3329	108	0.9686
	Négatif	135	3088	0.9581
		0.9610	0.9662	0.9635

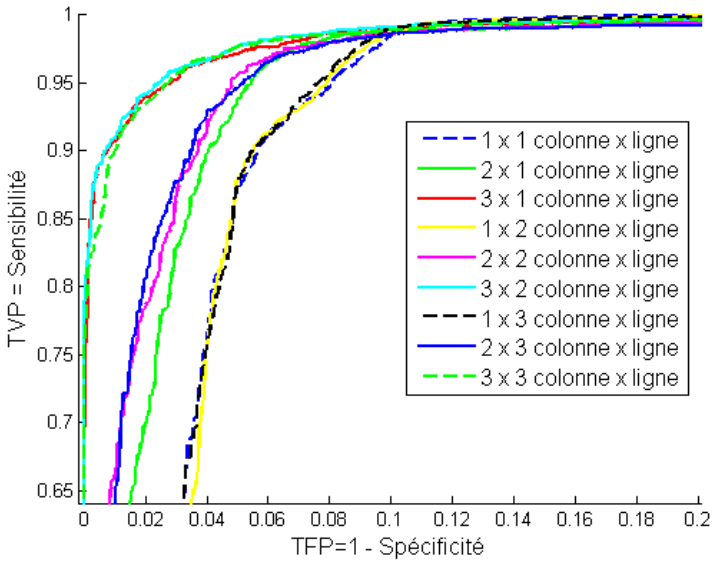


FIGURE 4.18 – Courbes ROC de détection pour différents descripteurs HOG.
En fonctions du nombre de blocs et leur disposition, le pouvoir discriminatoire de HOG varie. Les meilleures combinaisons sont 3×1, 3×2 et 3×3

Tableau 4.5 Matrice de confusion pour HOG 3×2

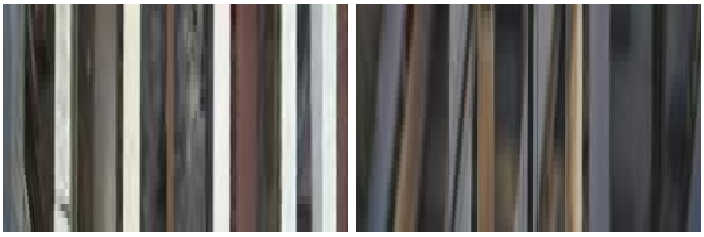
		Condition		
		Positive	Négative	
Test	Positif	3333	106	0.9692
	Négatif	131	3090	0.9593
		0.9622	0.9668	0.9644

Certains des faux négatifs se produisent au milieu de l'appareillage de voie, quand les rails traversent le centre de l'aiguille. Ceci explique que le résultat du *template matching* soit plus faible et en même temps les bords des rails font varier les orientations du descripteur de HOG. La figure 4.19 montre des exemples des erreurs de détection.

Cependant, la plupart des faux positifs sont isolés, ce qui les rend faciles à détecter. Un simple filtrage dans le temps permettra les supprimer. Aussi, nous avons étiqueté les images contenant le rail de guidage en station comme négatifs. Toutefois, le système le détecte comme un rail, ce qui produit de nombreux faux positifs. Un exemple de ce type de faux positif est montré dans la figure 4.19a. Ce type de faux positif est un cas à traiter à part, soit en utilisant la localisation, soit en recherchant la présence d'un quai.

En revanche, le seuil de probabilité du classifieur SVM peut être positionné pour avoir un rappel de 100 % au détriment de la précision et le système fournira en guise de réponse des nombreuses détections erronées en plus de celles pertinentes, il sera donc difficilement exploitable. Cependant, grâce à un filtrage temporel, ou même l'usage de la localisation ou de la détection des feux de signalisation permettra de réaliser une suppression des faux positifs et par conséquent augmenter la précision. Tous les croisements sont équipés des feux de signalisation, donc une combinaison de la détection des feux semble cohérent pour améliorer le taux de détection des croissements

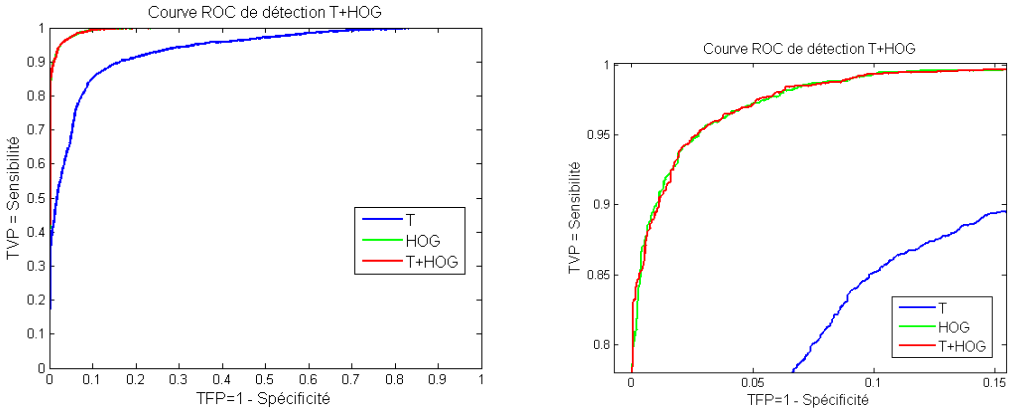
Un dernier test a été réalisé en combinant les deux descripteurs [Song *et al.* 2008b] mais comme le montre la figure 4.20, les résultats obtenus ne semblent pas beaucoup plus efficaces que HOG. Cela confirme que l'apparence et la forme locale d'un objet dans une image peuvent être décrites



(a) Faux positif de détection. (b) Faux négatif de détection.

FIGURE 4.19 – Erreurs de détection d'aiguille

par la distribution de l'intensité du gradient ou la direction des contours.



(a) ROC comparative de template matching, HOG 3×1, et leur combinaison.

(b) Zoom de la figure (a).

FIGURE 4.20 – Évaluation de l'apport simultané des deux descripteurs.

Les travaux réalisés dans ce domaine ont donné lieu à une publication internationale dans IEEE ICIRT 2013 *International IEEE Conference on Intelligent Rail Transportation* intitulé *Rail and Turnout detection using gradient information and template matching* (voir Annexe A.2).

4.5 Méthode proposée pour la classification du type d'appareil de voie

Pour la classification du type d'appareil de voie, la base de données est la même que précédemment, par contre, cette fois-ci, seules les images contenant des appareils de voie ont été gardées. Les différents types des appareils de voie présents dans les systèmes VAL, peuvent être réduits à 5 classes. Deux divergents, deux convergents et un pour l'arrivée et sortie de l'appareil de voie.

Le nombre total des images IMP avec appareils de voie est de 4949 où 2179 sont de la classe 1, 1100 sont de la classe 2, 344 sont de la classe 3, 730 sont de la classe 4 et 596 sont de la classe 5.

Vu le faible nombre d'échantillons, pour l'apprentissage supervisé nous avons utilisé seulement 30% de la base de données pour chaque classe, ceci pour éviter un surentraînement (le modèle colle parfaitement aux données) et un détriment des précisions futures.

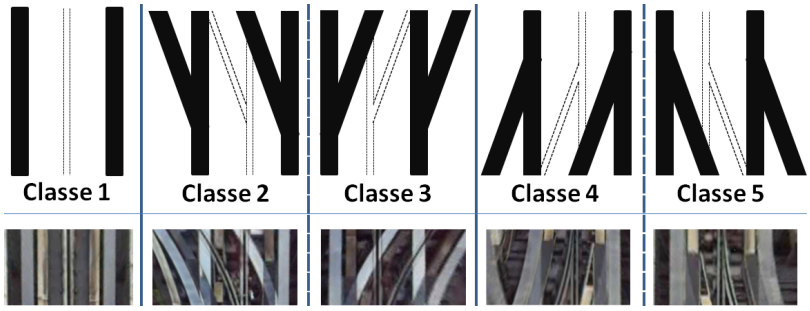


FIGURE 4.21 – Images IPM pour les 5 classes d'aiguillage

4.5.1 SVM multi-classe

À l'origine, les SVMs ont été conçus essentiellement pour la séparation de 2 classes. Cependant, plusieurs approches permettent d'étendre cet algorithme aux cas de plusieurs classes. Ces schémas sont applicables à tout classificateur binaire, et ne sont donc pas spécifiques des SVM [Hsu & Lin 2002].

Un Contre Tous (1vsA : *One versus All)*

Pour chaque classe, il faut déterminer un hyperplan séparant celle-ci de toutes les autres classes. Ainsi, pour M classes, le nombre total de fonctions de décision est M.

Pour chaque fonction de décision, tous les exemples appartenant à la classe considérée sont étiquetés positivement (+1) et tous les exemples n'appartenant pas à la classe sont étiquetés négativement (-1). En phase de test, le classificateur donnant la valeur de confiance (e.g la marge) la plus élevée remporte le vote.

Ainsi, pour chaque exemple de test, M valeurs de sortie $fi(x)$ avec $(i = 1, ...M)$ sont disponibles.

Un contre un (1vs1 : *One versus One)*

Pour cette deuxième méthode, au lieu d'apprendre M fonctions de décisions pour différencier les classes, ici chaque classe est discriminée d'une autre. Ainsi $\frac{M(M-1)}{2}$ fonctions de décisions sont nécessaires et chacune d'entre elles effectue un vote pour l'affectation d'un nouveau point x. Pour chaque fonction de décision, deux classes sont comparées. Tous les exemples appartenant à la classe considérée sont étiquetés positivement (+1) et tous les exemples appartenant à l'autre classe sont étiquetés négativement (-1).

En phase de test, l'échantillon à classer est analysé par chaque classificateur et un vote majoritaire permet de déterminer sa classe.

4.5.2 Descripteur proposé : HOG

Puisque le descripteur de HOG est caractérisé par sa grande capacité à discriminer les formes, en particulier les piétons [Dalal & Triggs 2005] et les panneaux routiers [Zaklouta *et al.* 2011], nous avons décidé d'utiliser ce descripteur pour réaliser notre classification des appareils de voie.

Tests réalisés avec Pyramide d'Histogrammes orientés de Gradients (PHOG)

[Bosch *et al.* 2007] nous présente une variante de HOG, le PHOG (de l'anglais *Pyramid of Histograms of Oriented Gradients*. Ce descripteur est présenté comme plus précis que le HOG puisqu'il lui apporte un aspect spatial supplémentaire, l'obtention des HOG à différentes échelles. La figure 4.22 montre le calcul du PHOG sur les différents niveaux de décomposition, en particulier pour le niveau de décomposition $L = 2$.

De plus, selon son créateur [Bosch *et al.* 2007], PHOG montre de très bons résultats de reconnaissance des formes. Son efficacité a été prouvé dans les travaux de [Puthon 2013] pour la classification des panonceaux routiers en 4 groupes : «Texte» «Flèche» «Pictogramme» et «Mixte».

L'obtention de PHOG peut être décrite comme suit :

Au niveau l, l'image est subdivisée en 4^l sous-régions. Dans chacune d'elle, l'image des contours est synthétisée en un histogramme de de B intervalles d'orientation de gradients pondérés par leur amplitude. Ces intervalles sont définis de la même façon que pour HOG, B conteneurs pour des gradients signés ou non signés. Le descripteur global correspond à la concaténation des vecteurs obtenus pour chaque niveau, soit une dimension totale $B \sum_{l \in L} 4^l$. Enfin, une normalisation est effectuée sur le vecteur complet. Contrairement à l'approche proposée par [Dalal & Triggs 2005] pour la détection de piétons, il n'y a pas de recouvrement des blocs, ni de partitionnement plus fin en cellules ni de normalisation par blocs.

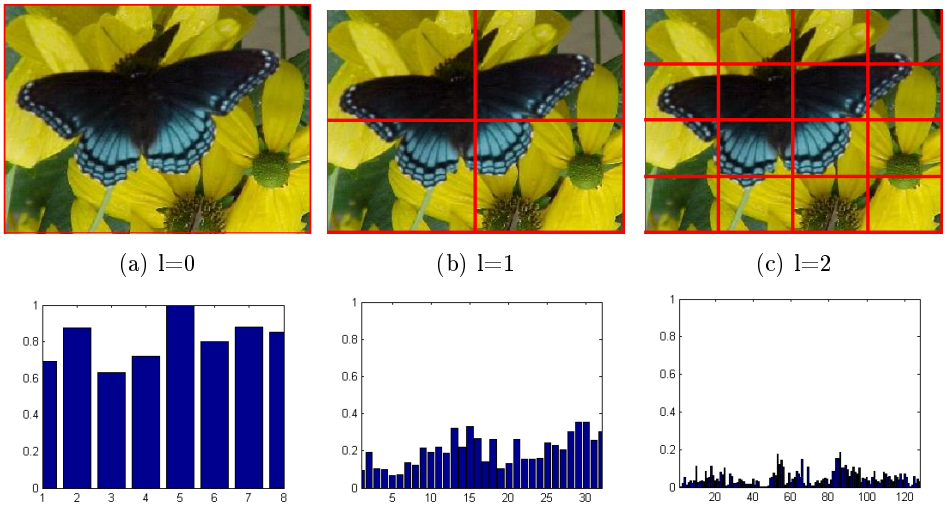


FIGURE 4.22 – Génération d'un PHOG pour une image. $B = 8$, $L = 3$.
À chaque niveau l, l'image est décomposée en 4 sous-régions pour lesquelles un histogramme d'orientation de gradients sur B intervalles est obtenu. La concaténation de tous les vecteurs de chaque niveau forme le descripteur final. La normalisation est réalisée pour l'ensemble du vecteur.

Puisque nos images IPM ont une taille réduite, 32×256 *pixels*, nous avons limité le niveau maximal de PHOG à L=3. Des niveaux de division en blocs supplémentaires nous donneraient des descripteurs plus grands que la l'image d'entrée elle-même. Le nombre d'intervalles a été fixé à 8 pour des gradients non-signés. L'usage des gradients signés n'apportera pas d'information supplémentaire pour différencier nos 5 types des classes.

Les premières démarches de classification avec PHOG nous ont donné de bons résultats comme le montre la figure 4.23, où nous montrons l'exactitude globale du classifieur multi-classe pour deux méthodes, SVM $1vs1$ et $1vsA$.

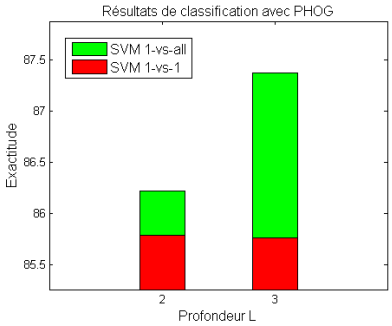


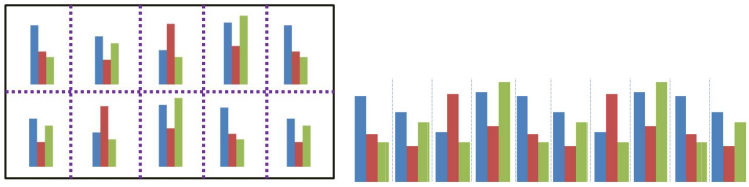
FIGURE 4.23 – Résultats de classification pour le descripteur PHOG. .
Pour cette classification, le nombre de *bins* utilisé est de 8 avec des gradients non-signés. La barre rouge nous montre les résultats avec un classifieur SVM 1vs1 et la verte SVM 1vsAll. L'abscisse représente le degré de profondeur de PHOG.

Vu les bons résultats obtenus avec PHOG, nous avons considéré que nos images IPM étant toujours vues depuis le même point, nous pouvons mieux cibler le positionnement des blocs de HOG pour différencier les classes. Contrairement au découpage de PHOG, nous proposons de réaliser un découpage classique de HOG, sans chevauchement.

Cette fois, les conteneurs (*bins*) de HOG, ont été définis comme indiqué dans l'équation 4.11, un total de 10 *bins*, avec l'objectif de mettre en valeur la présence des rails convergents et divergents.

$$\begin{cases} [0-10;\ 10-30;\ 30-50;\ 50-70;\ 70-85;\ 85-95; \\ 95-110;\ 110-130;\ 130-150;\ 150-170;\ 170-180] \end{cases} \tag{4.11}$$

Pour déterminer le nombre de colonnes/lignes de HOG, nous avons testé différentes combinaisons. Un exemple de partition est montré dans la figure 4.24, pour HOG 5×2.



(a) Division en blocs de l'image IPM.

(b) Descripteur HOG (5 colonnes $\times$ 2 lignes).

FIGURE 4.24 – Division en blocs de l'image IPM et concaténation des histogrammes pour former le descripteur.

4.5.3 Résultats de classification

Pour déterminer le bon nombre de blocs, nous avons testé différentes combinaisons des colonnes/lignes, et leur résultat d'exactitude globale est montré dans la figure 4.25. Nous pouvons voir qu'à partir d'un certain nombre de colonnes, le résultat de classification plafonne et se voit même réduit. Le meilleur descripteur est 7 colonnes et 3 lignes.

Pour mieux analyser les performances du classifieur, nous examinons sa réponse individuelle pour chaque classe. Pour cela, nous représentons la courbe ROC Précision-Rappel. Cette courbe nous permet de mettre en évidence deux aspects de la classification :

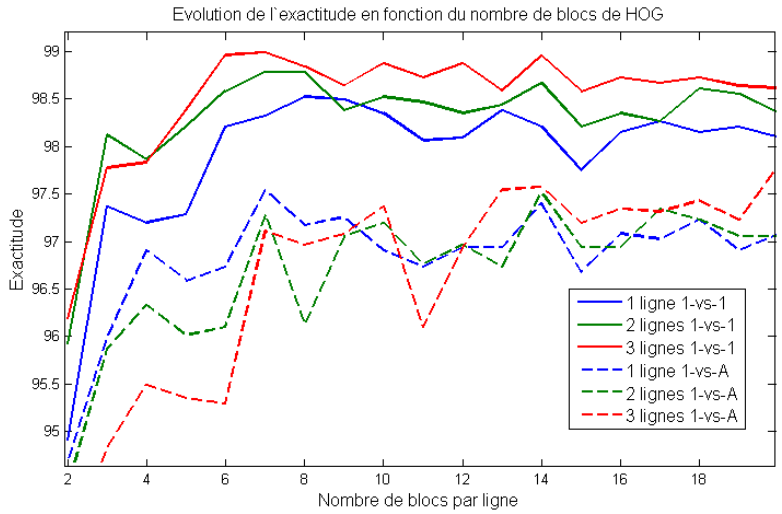


FIGURE 4.25 – Élection du nombre de lignes et de colonnes pour le descripteur de HOG. Nous pouvons voir comment varie l'exactitude pour les deux classifieurs SVS $1vs1$ et SVM $1vsA$ en fonction du nombre de colonnes de lignes du descripteur HOG. SVM $1vs1$ se présente plus efficace que $1vsA$ pour tout les cas. La meilleure exactitude est obtenue pour 7 colonnes et 2 lignes. La taille finale du meilleur descripteur est de $7 \times 3 \times 10 = 210$. L'exactitude est de 98.98%

- Le taux de précision ou la proportion des vrais positifs dans les images testées
- les taux de rappel ou la proportion de classes bien déterminées parmi toutes les images présentes dans la base de données.

Les courbes ROC de la figure 4.26 ont été obtenues en faisant varier le seuil de probabilité d'appartenir aux positifs pour chacun des SVM du classifieur multi-classe. Plus le seuil est haut, meilleure est la précision et pire le rappel.

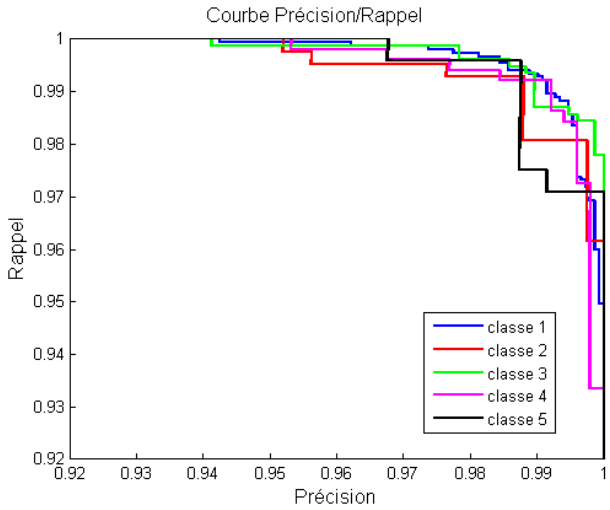


FIGURE 4.26 – Courbe Précision-Rappel obtenue pour SVM $1vs1$ avec descripteur HOG 7×3 pour chacune des classes.

La figure 4.28 montre différents exemples de classification des appareils de voie.

Afin d'évaluer quantitativement les erreurs de confusion entre classes, nous analysons la matrice de confusion. Pour le problème donné à $C = 5$ classes, la matrice de confusion obtenue dispose de $C^2 = 25$ positions, contenant les C bonnes classifications sur la diagonale principale, et $C(C-1)$

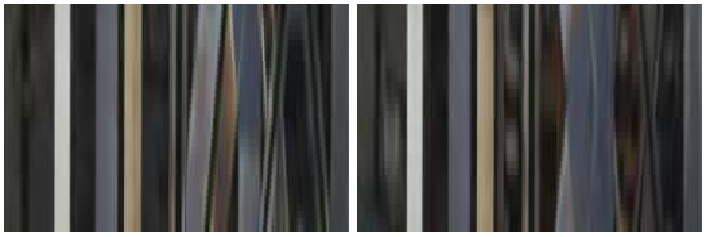


FIGURE 4.27 – Erreurs de classification.

erreurs possibles hors de la diagonale. Pour le meilleur classifieur SVM $1vs1$ et un descripteur de HOG 7×3 la matrice de confusion est le tableau 4.6 expose les performances.

Tableau 4.6 Matrice de confusion pour la classification des aiguilles

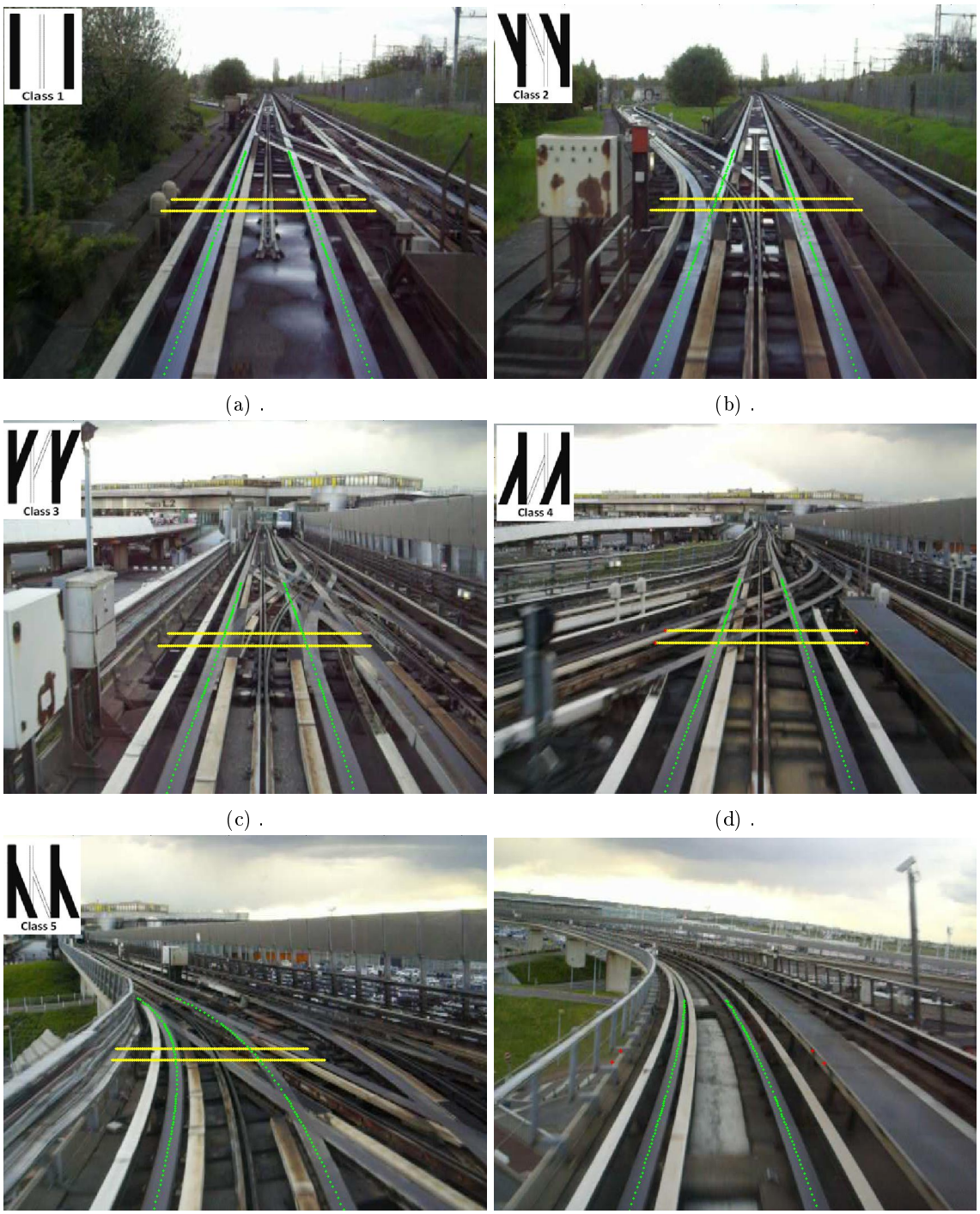
		Classifieur					
		c.1	c.2	c.3	c.4	c.5	Rappel
Vérité terrain	c.1	1511	6	0	1	7	99.08%
	c.2	6	762	0	2	0	98.96%
	c.3	2	0	235	2	1	97.92%
	c.4	4	0	1	505	0	99.02%
	c.5	1	0	2	0	414	99.28%
	Précision	99.15%	99.22%	98.74%	99.02%	98.10%	98.98%

La plupart des erreurs de classification interclasse se produisent dans les croisements croisés (double communication entre 2 voies). Dans l'image IMP, les deux voies sont visibles et pourtant le classifieur a du mal à prendre une décision. Un exemple est montré dans la figure 4.27.

De très bons résultats préliminaires ont été obtenus pour les différentes classes. Le rappel minimum atteint est de 97.92% et la précision 98.10% sur un ensemble de 3462 images avec appareils de voie.

L'approche proposée pour la classification des appareils de voie pourrait aussi s'utiliser pour les appareils de voie ferrée traditionnelle, car la présence de la nouvelle voie introduit également des variations dans les gradients de l'image.

Ces travaux de classification d'appareils de voie ont donné lieu à une publication internationale dans IEEE ITSC 2013 *16th International IEEE Conference on Intelligent Transportation Systems* intitulée *Turnout detection and classification using modified HOG and template matching* (voir Annexe A.3).



(a) . (b) .

(c) . (d) .

(e) . (f) .

FIGURE 4.28 – Résultats de la classification d'aiguillage pour les différentes classes. *Classe* 1; *Classe* 2; *Classe* 3; *Classe* 4; *Classe* 5; et *Sans aiguille*

Chapitre 5

Détection d'obstacles : trains

Sommaire

5.1 Introduction **97**
5.1.1 Systèmes en commercialisation industrielle 99
5.2 Base de données **100**
5.3 Sélection du descripteur **102**
5.3.1 Descripteurs locaux 102
5.3.2 Analyse globale : Histogramme de gradients orientés (HOG) 106
5.4 Résultats **107**

5.1 Introduction

En ce qui concerne la localisation et détection d'obstacles sur la voie ferrée, il existe plusieurs approches avec différents types de capteurs comme illustré dans le chapitre 2. De plus, nous avons pu les classifier en deux grands groupes : ceux équipés de capteurs passifs tels que les systèmes de vision et ceux équipés de capteurs actifs tels que les radars et les télémètres lasers.

D'autre part, le critère de détection d'obstacles dépend de leur nature. La définition la plus générale est de considérer comme obstacle n'importe quel objet qui gêne le chemin du véhicule, cependant ceci induit une plus grande complexité algorithmique. La deuxième se limite à considérer comme obstacles les autres véhicules et, puisque leur définition est bien définie, il est alors possible de faire une recherche des patrons spécifiques comme la forme et la symétrie.

Enfin, indépendamment de la définition de l'obstacle, deux approches de détection grâce à la vision par ordinateur sont envisageables : l'approche mono caméra et l'approche multi caméra.

Les approches mono-caméra procèdent à l'analyse de certaines primitives extraites dans plusieurs images consécutives. Ces primitives peuvent être des régions texturées, des points de contour ou des zones en mouvement. La perte de l'information de profondeur est la première conséquence de la projection du plan d'un monde tridimensionnel sur un capteur 2D (plan image). Ces méthodes ont systématiquement recours à des connaissances *a priori* sur l'environnement ou sur la typologie et la géométrie des obstacles à détecter.

Les approches multi-caméra exploitent la segmentation d'une carte des disparités calculées à partir de l'analyse des images fournies par un stéréoscope étalonné, comme c'est le cas de la V-disparité, une représentation proposée par [Labayrade *et al.* 2002]. Cette représentation exploite une hypothèse de monde plan, et représente l'accumulation de la disparité calculée sur chacune des lignes de l'image. Les cartes des disparités sont généralement établies sur plusieurs trames successives afin de garantir une cadence raisonnable des traitements et ainsi respecter la contrainte du temps réel requise pour un organe de sécurité. Ainsi, [Dooze *et al.* 1998, Louahdi *et al.* 2002] nous présentent des projets de détection d'obstacles sur la voie ferrée développés à partir de stéréovision.

De plus, en vision artificielle et traitement d'images, les systèmes de détection d'objets exigent, en général, des algorithmes complexes. Une recherche exhaustive dans une image complète peut-

être envisagée pour des applications de type *off-line* ou d'autres, où le temps réel n'est pas une contrainte. Les systèmes de détection des obstacles embarqués, de leur côté, doivent être conçus pour évaluer leur environnement à une vitesse telle que, dans le cas du déclenchement d'une alarme, le système puisse avoir un temps de réaction compatible avec les performances demandées.

De nombreux travaux ont été menés dans le cadre du développement des aides à la conduite, et tout particulièrement de la fonction de détection et de suivi des obstacles. Après avoir analysé plusieurs d'entre eux, nous pouvons établir le tableau 5.1 montrant quelques avantages et inconvénients [Bertozzi *et al.* 2000].

Tableau 5.1 Comparatif des méthodes pour la détection d'obstacles.

Technique	Avantages	Inconvénients
Analyse d'une seule image	Solution algorithmique simple et rapide. Indépendance des mouvements du véhicule.	Manque d'information de profondeur si on n'a pas fait d'hypothèse. Inefficace quand les obstacles ne correspondent pas au modèle.
Flux optique (Ego motion)	Détection des obstacles génériques.	Exigeant en puissance de calculs. Sensible aux mouvements du véhicule. Échec si l'obstacle et le véhicule ont une vitesse faible.
Stéréo vision	Détection des obstacles génériques. Permet une reconstruction 3D.	Très exigeant en puissance de calculs. Sensible aux mouvements du véhicule et dégradation de l'étalonnage.

En se basant sur le tableau 5.1 et le fait que notre dispositif a été défini en monovision, nous proposons la technique d'analyse image par image pour détecter une rame sur la voie.

Comme nous travaillons avec des systèmes dans un site propre intégral, les obstacles potentiels à détecter seront les autres rames, ce qui nous permettra de faire une recherche des patrons spécifiques comme la forme et la symétrie.

La plupart des systèmes de détection des véhicules par monovision travaillent généralement en deux étapes : la génération de l'hypothèse et la vérification de l'hypothèse. La première consiste en une recherche rapide à travers l'image afin de déterminer des portions plus petites susceptibles d'être des véhicules. Cette recherche est principalement basée sur certaines caractéristiques que pourront avoir les véhicules, telles que la couleur [Tsai *et al.* 2007], l'ombre [Tzomakas & von Seelen 1998], des gradients verticaux [Song *et al.* 2008a] ou le flux des images successives [Jazayeri *et al.* 2011]. Si une technique d'exploration basée sur des fenêtres glissantes est utilisée, le nombre des fenêtres à évaluer peut être énorme.

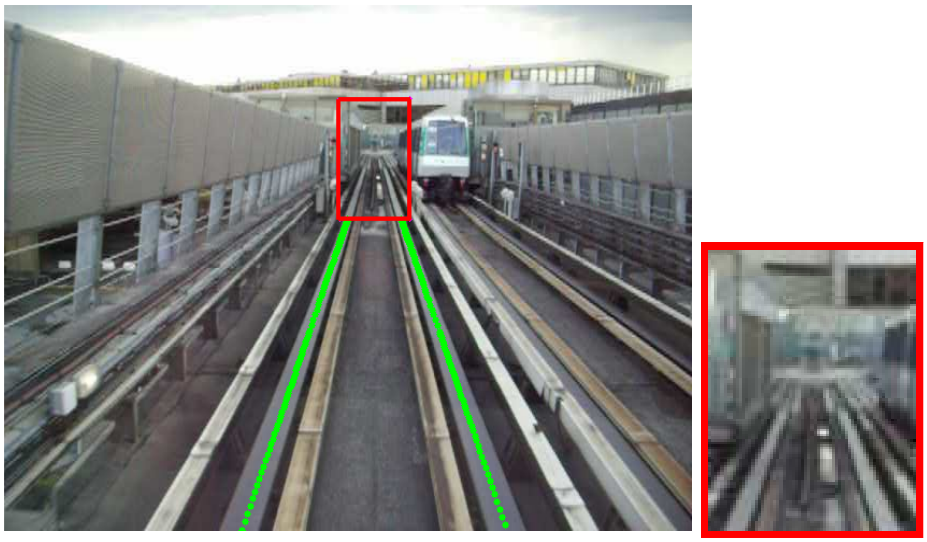
La seconde étape, la présence des véhicules, est testée pour différentes tailles, et généralement est réalisée avec un classifieur binaire, lequel a été entraîné avec des images de véhicules et de non-véhicules.

Comme nous nous servons de la détection de la voie, la première étape se voit simplifiée, seules les parties de l'image où la détection de la voie n'arrive plus à distinguer les rails seront considérées comme obstacles. Le fait de ne pas réaliser des recherches exhaustives à travers l'image en cherchant des indices des trains à toutes les positions et échelles possibles (*sliding window* en anglais), entraîne une réduction du temps de calcul et comme moins d'hypothèses sont testées. La probabilité de tomber sur des fausses détections est plus faible.

Certains travaux utilisent différentes primitives pour définir les zones d'intérêt (ROI) [Xavier 2003]

où la détection des véhicules routiers combine un module de détection des voies, la reconstruction de la géométrie 3D de la route, un module de détection de l'ombre et un module de localisation de la symétrie.

Parce que notre système est une version simplifiée de la détection des véhicules routiers grâce à l'extraction de la voie, nous pourrons définir une zone libre d'obstacles et la région d'intérêt pour laquelle nous allons vérifier l'hypothèse en déclenchant une recherche des trains. Cette ROI, sera simplement un rectangle avec la taille d'un train à la distance maximale d'extraction des rails, en utilisant les lois de projections de l'équation 2.2.



(a) Extraction de voie et détermination de la ROI. (b) ROI.

FIGURE 5.1 – Détermination de la ROI à partir de l'extraction de rails.

La figure 5.1 montre un exemple de la ROI définie. Le fait d'utiliser l'extraction de voie courante pour définir la ROI nous permet d'éviter d'avoir des fausses alarmes (faux positifs) à cause des détections des trains d'une voie parallèle. En outre, la seule différence visuelle entre deux trains qui circulent en direction opposée est la couleur des phares de la rame.

5.1.1 Systèmes en commercialisation industrielle

Les systèmes de prévention de collisions développés dans le domaine automobile sont basés principalement sur l'utilisation des capteurs actifs comme le RADAR ou le LIDAR, parfois associés à une caméra. Ces capteurs actifs ont été décrits plus en détail dans la section 2.2.Un résumé des principales caractéristiques et actions menées par ces systèmes est montré dans le tableau 5.2. Ces données ont été extraites du *European New Car Assessment Program (Euro NCAP)*, c'est-à-dire le «programme d'évaluation européenne des automobiles»,[NCAP 2013].

Tableau 5.2 Systèmes industriels de détection d'obstacles en commercialisation.

Constructeur	Application	Équipement	Vitesse (km/h)	Actions
Volvo	Ville	Radar et caméra	Jusqu'à 35	Freinage d'urgence (FU)
Ford	Ville	Laser et caméra	Jusqu'à 30	FU
Honda	Ville et autoroute	Radar	À partir de 15	3 phases : Alarme, préparation des freins et FU
Mercedes	Autoroute	3 Radars	Entre 30-200	2 phases : Alarme et FU

Dans tous les cas, les constructeurs d'automobiles privilégient logiquement «l'inaction» sur l'action puisqu'il s'agit d'assistance à la conduite et que leur responsabilité pourrait être engagée en cas d'accident dû à un freinage d'urgence intempestif.

Cependant, à notre connaissance, aucun système de détection d'obstacles n'est opérationnel sur un métro entièrement automatique.

5.2 Base de données

Pour pouvoir réaliser la reconnaissance des trains, il est nécessaire de développer un classifieur avec un apprentissage supervisé, donc nous avons besoin d'une base de données avec des exemples positifs et négatifs. Pour obtenir les images du groupe des négatifs, nous avons utilisé la base de données publique VOC2012 [Everingham *et al.*] spécialement conçue pour la détection et reconnaissance des objets. Il existe 4 types de grandes classes : personnes, animaux, véhicules et objets d'intérieur. Cependant, nous avons seulement utilisé la classe «véhicule» en particulier les images des voitures et des bus, car sous certains points de vue, leur forme peut ressembler à un train. Pour avoir des images plus détaillées et éviter de tenir compte de l'environnement de l'image, nous avons choisi d'extraire les voitures et les bus grâce à la vérité terrain de la base de données. Un exemple de cette extraction est montré dans la figure 5.2.

Ainsi, pour incrémenter le nombre d'images négatives, nous avons utilisé la base d'images de voitures de *l'INRIA car data set* (utilisée dans les travaux de [Carbonetto *et al.* 2008]) ainsi que des images prises sur différents sites web. Ces dernières ont été ajoutées avec l'objectif de donner encore plus de variabilité aux négatifs et comprennent principalement des images de portes, de fenêtres, de voitures, de bus et même d'autres types de trains. La figure 5.4 montre quelques exemples des images utilisées pour entraîner et tester notre algorithme d'apprentissage.

Les positifs de la base de données sont des images du métro VAL dans ses différentes versions. Principalement le métro est vu de face, mais certaines images ont une orientation différente, comme cela pourrait arriver dans un cas réel. La figure 5.3 montre les images du VAL utilisées. L'origine de ces images est très diverse, principalement ce sont des images obtenues lors de nos acquisitions à bord de l'Orly VAL et à Rennes. Cependant, d'autres appartiennent à des vidéos ou des photos d'Internet. Ceci contribue à avoir une diversité dans les images, car elles ont été obtenues avec différents appareils et sous différentes conditions d'éclairage.

La base de données compte, au total 884 positifs et 896 négatifs, en générant des images symétriques verticales à partir des images d'origine, en anglais *flopped images*.

(a) Image d'origine. (b) Bus extrait. (c) Image d'origine. (d) Voiture extraite.

FIGURE 5.2 – Obtention de la base de données à partir de VOC2012 [Everingham *et al.*].

FIGURE 5.3 – Images des trains «positifs».

FIGURE 5.4 – Images des «négatifs».

5.3 Sélection du descripteur

Les méthodes de classification ont pour but d'identifier les classes auxquelles appartiennent des objets à partir de certains traits descriptifs, et dans cette section, nous allons sélectionner le meilleur descripteur qui s'adapte à notre application. Ce choix consiste à identifier les variables pertinentes, car si la sélection des variables est réalisée de manière appropriée, la performance du modèle de discrimination peut être fortement améliorée.

Nous pouvons distinguer deux façons d'extraire ces caractéristiques, les caractéristiques globales qui sont calculées sur toute l'image et les caractéristiques locales qui sont calculées autour de points d'intérêt.

Les caractéristiques locales ou points d'intérêt sont distinctes, robustes aux occlusions (car il y en a beaucoup dans une image ou une région) et qu'elles ne nécessitent pas de segmentation.

Les caractéristiques globales utilisent l'ensemble de l'image, comme la texture, la couleur, les histogrammes. Le plus connu est l'histogramme de Gradients Orientés (HOG). Cependant, aucun moyen ne permet de distinguer l'objet de son fond, car les données utilisées appartiennent aux deux catégories.

Par ailleurs, le fait de réduire le nombre de variables permet également de réduire le temps de calcul, aspect non négligeable pour un système temps réel.

5.3.1 Descripteurs locaux

La détection de points d'intérêt dans les images est de plus en plus utilisée afin de faciliter de nombreuses tâches comme la reconnaissance d'objets, l'assemblage d'images, la modélisation 3D, la recherche d'image par le contenu et même le *tracking* video.

De façon générale, les travaux sur la détection des points d'intérêt ont commencé à partir des travaux de [Harris & Stephens 1988] pour le développement d'un détecteur de coins et de contours, suivis par le de détecteur de *blobs* (petites régions uniformes qui se démarquent) comme le SIFT de l'anglais *Scale-Invariant Features* introduit par [Lowe 1999] et finalement les SURFs de l'anglais *Speeded-Up Robust Features* proposés par [Bay *et al.* 2006], une variante plus rapide de SIFT.

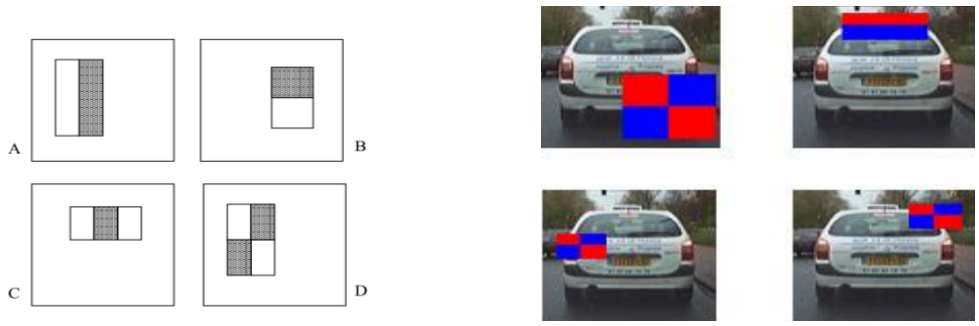
Ces algorithmes cherchent à isoler les régions de l'image qui présentent des caractéristiques visuelles remarquables, comme les bords, les coins et les *blobs* de l'image, et leur qualité est mesurée en fonction de la capacité à retrouver les mêmes points d'intérêt d'un objet dans une scène lorsque les conditions de visualisation changent : invariants à un changement d'échelle, à la rotation, à la perspective ou au changement léger de point de vue.

SIFT est un détecteur de points d'intérêt invariant aux changements d'échelle combiné avec un descripteur basé sur la distribution des gradients calculés sur la région d'intérêt. Quant à SURF, il se base sur le principe des images intégrales [Viola & Jones 2001] pour accélérer les calculs et décrit la distribution des réponses à une ondelette de Haar dans le voisinage du point d'intérêt. La figure 5.8 montre l'extraction des SURFs pour des images ferroviaires. L'image intégrale est une représentation sous la forme d'une image numérique et un algorithme, permettant de calculer rapidement des sommes de valeurs dans des zones rectangulaires.

Ainsi, ces types de descripteurs sont très efficaces pour la détection et le suivi temporel des obstacles comme le montrent les travaux de [Lefaudeux 2013] grâce à la stéréovision, et même pour la ré-identification des personnes avec différentes caméras [Hamdoun 2010].

Viola et Jones, ont proposé *Haar-like features* un descripteur et un classifieur pour la détection des visages [Viola & Jones 2001]. Leur approche est basée sur une variation des ondelettes de Haar obtenues par la différence entre deux régions d'un modèle rectangulaire comme le montre la figure 5.5a. Ces caractéristiques se calculent rapidement grâce à la propriété des images intégrales. L'extraction des caractéristique Haar-like s'obtient en faisant la différence entre la somme des pixels de

la zone blanche et de la somme de la zone foncée. Ensuite les meilleures caractéristiques sont sélectionnées par un algorithme de *boosting*. Ceci consiste à assembler plusieurs classifieurs faibles afin d'obtenir un classifieur à forte capacité. De plus, ils utilisent une variante d'un algorithme d'AdaBoost en mode cascade [Barczak *et al.* 2008]. La configuration en cascade offre une grande économie en temps de calcul, car les premiers classifieurs permettront de filtrer le plus grand nombre de négatifs sans besoin de tester tous les classifieurs faibles. Les cascades ayant des limitations, comme la détermination empirique des bons taux des faux positifs à chaque étape, leur temps d'entrainement est lent, et finalement l'utilisation des cascades peut réduire les performances globales du système [Harzallah 2011].



(a) Exemple des masques de rectangles tirés de [Viola & Jones 2001].

(b) Rectangles Haar positionnés dans l'image

FIGURE 5.5 – Exemple de l'extraction des caractéristiques de Haar

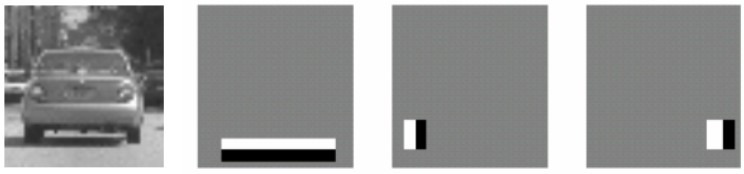
La figure 5.6a nous montre les principales caractéristiques de Haar retenues par le *boosting* après un apprentissage sur des voitures [Lee 2007]. La figure 5.6b nous montre une variation des caractéristiques de Haar, basée dans les *Control Points*[Abramson *et al.* 2007] sont les *Connected Control Points*. Les *Control Points* sont constitués de deux groupes de points (positifs et négatifs), où un exemple est classé positif, si la différence entre la valeur minimale représentée par un pixel positif es la valeur maximale d'un pixel négatif(et inversement) est supérieure à un seuil. L'idée des *Connected Control Points* est de réduire le temps de calcul et d'ajouter des contraintes de symétrie et topologiques aux caractéristiques, a consisté d'ajouter une contrainte de connexité sur un voisinage de 8, ce qui implique que chaque point doit toucher au moins un autre par un coin. Ces travaux nous présentent des taux de détection de voitures aux alentours de 90 %.

Un des exemples d'utilisation des points d'intérêt pour la détection des voitures vues de profil est présenté par [Bdiri *et al.* 2009]. Il nous propose une variante de SURF que nous pouvons trouver dans la libraire CAMELLIA [Steux 2001], et un classifieur *Adabosting* pour déterminer les points d'intérêt appartenant à des voitures, suivi d'une analyse basée sûr Hough, pour déterminer la position des points et décider si l'ensemble des points détectés forment une voiture. Les résultats atteignent une précision de 95%. La figure 5.7 nous montre un exemple.

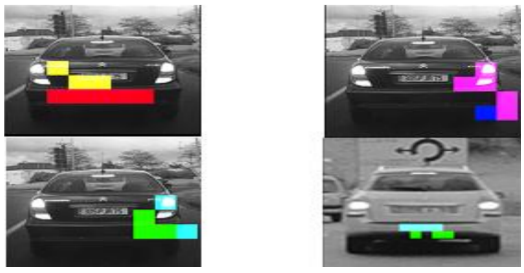
Or, dans des contextes perturbés et notamment bruités, les performances de ces descripteurs locaux restent faibles par rapport aux descripteurs globaux qui travaillent sur l'image entière. De plus, le nombre de points varie d'une image à une autre, et nécessite une bonne résolution de départ sur l'image, ce qui dans un environnement externe et pour des grandes distances n'est pas toujours possible. L'aspect spatial est également perdu et la position relative des points n'est pas conservée.

L'utilisation des points d'intérêt implique d'avoir un grand nombre de descripteurs, ce qui peut conduire à de faux appariements et par conséquent à de fausses détections. De plus, nous ne pouvons pas garantir que les points d'intérêt soient toujours visibles dans toutes les conditions d'éclairage à la distance de détection que nous voulons atteindre.

Ces raisons nous obligent à écarter ce type de descripteurs pour l'utilisation de descripteurs



(a) Meilleures caractéristiques de Haar pour la détection des voitures.



(b) *N-Connexity Control Points* retenus pour la détection des voitures.

FIGURE 5.6 – Images extraites de [Stanciulescu *et al.* 2009] et [Lee 2007].

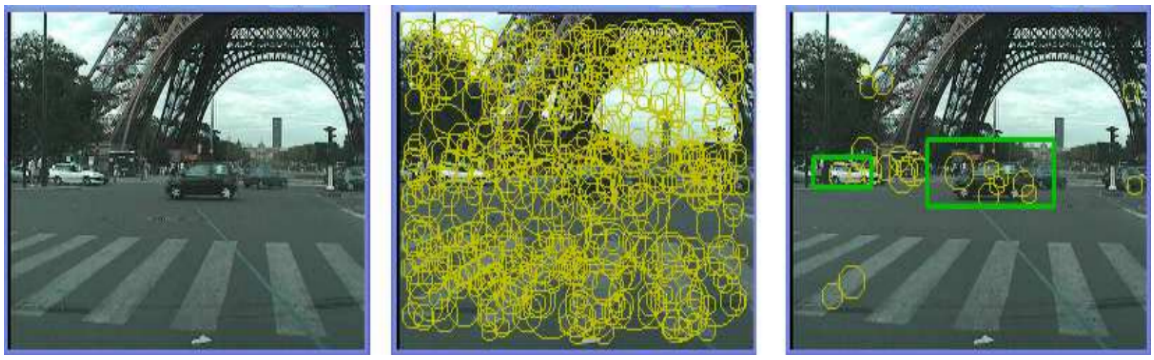
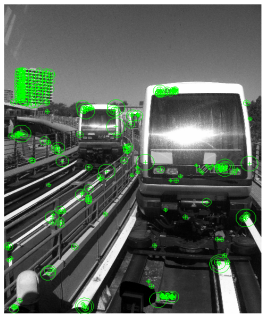


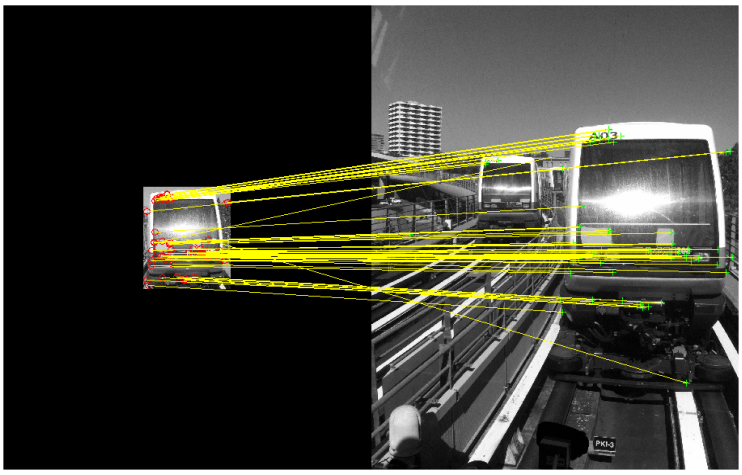
FIGURE 5.7 – Détection des voitures à partir des points d'intérêt. [Bdiri *et al.* 2009]

L'image de gauche montre l'image originale. L'image centrale montre les extractions des points d'intérêt, et l'image de droite les points d'intérêt rétenus comme des points appartenant à des voitures et les détections des voitures après avoir analysé la position des points grâce à une méthode semblable à la transformée de Hough.

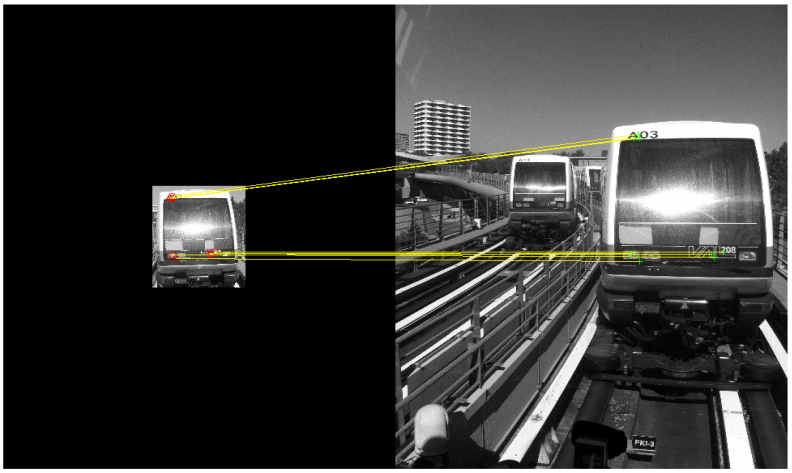
(a) Points d'intérêt SURF pour le train



(b) Points d'intérêt pour la scène.



(c) Appariement entre points d'intérêt de deux images.



(d) Correspondance des points après appariement avec RANSAC pour éliminer les faux appariements.

FIGURE 5.8 – Extraction des points SURF pour la détection d'obstacles.

Cette image nous montre l'usage des points d'intérêt pour la détection d'obstacles. D'abord, il faut caractériser l'objet par des points SURF (subfigure (a)). En second lieu, il faut extraire les points d'intérêt dans la scène et trouver des points ressemblants entre la scène et l'objet (subifugre (b)et(c)). Grâce à la connaissance de la distribution spatiale des points dans l'image d'origine, nous pouvons éliminer les mauvais appariements (subfigure (d)). Nous pouvons voir qu'il existe un grand nombre de points qui ont été appariés au début, mais qui finalement ont été éliminés. En outre, certains des points d'intérêt se concentrent dans des petits détails du train, comme le numéro ou le modèle, qui sous certaines conditions (réflets, distance,...), ne seront toujours visibles.

globaux. Cependant, ils peuvent être utilisés dans des applications futures pour augmenter l'efficacité du descripteur global.

5.3.2 Analyse globale : Histogramme de gradients orientés (HOG)

Compte tenu de la nécessité d'un système temps réel dans le domaine des ITS, la majorité des descripteurs utilisés actuellement sont peu gourmands en puissance de calculs, comme la symétrie de la partie arrière des véhicules, par exemple [Broggi *et al.* 2004]. En revanche, [Arrospide *et al.* 2012] nous démontre que la symétrie est un descripteur possédant une capacité limitée pour discriminer les images avec véhicules des images sans. La figure 5.9 nous montre la distribution spatiale d'un descripteur de symétrie pour une base de données des voitures et non-voitures. Pour cette raison ils ont proposé d'utiliser un descripteur basé sur HOG, car la partie arrière des voitures se caractérise par des gradients verticaux et horizontaux (la plaque d'immatriculation, la vitre arrière, etc). Une analyse du descripteur de HOG, des amplitudes et des distances entre certains *bins* est réalisée pour obtenir un deuxième descripteur. Les résultats obtenus dans la classification atteignent en moyenne un 92.48% d'exactitude.

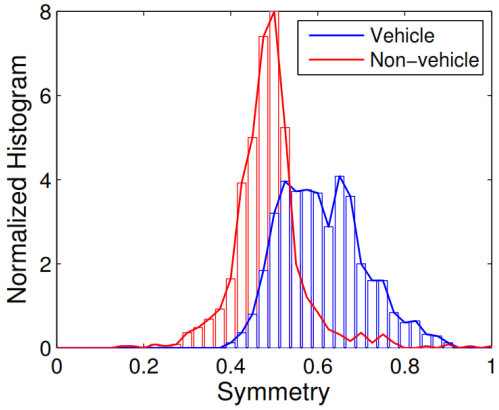


FIGURE 5.9 – Distribution du descripteur de symétrie testé par [Arrospide *et al.* 2012].

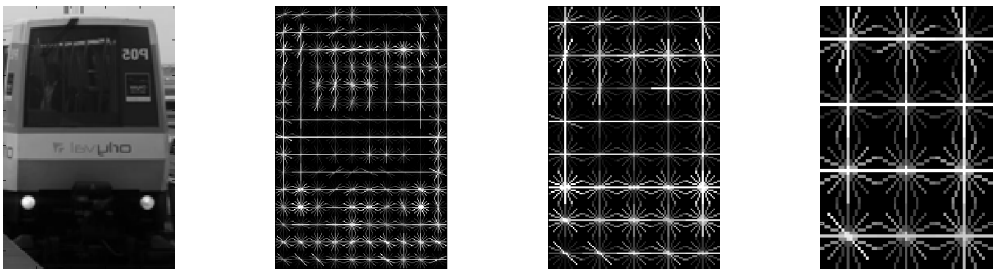
Vu les résultats obtenus dans les travaux de [Arrospide *et al.* 2012] pour la détection des voitures et les travaux de [Harzallah 2011] pour la classification multi-classe, nous proposons d'utiliser le descripteur de HOG pour la reconnaissance de la forme du train dans nos images ROI sélectionnées à partir de la détection de la voie. Si la voie n'est plus visible à cause d'un train, la silhouette du train peut être encadrée dans la ROI, ce qui nous permettra d'avoir un descripteur fixe et par conséquent une bonne distribution des blocs et cellules du descripteur de HOG, qui permettra de caractériser le profil du métro.

Pour obtenir le descripteur de HOG, il est nécessaire de fixer certains paramètres comme le nombre des blocs, des cellules, des *bins* et l'intervalle des orientations. La version de HOG utilisée, en termes de distribution des blocs/cellules est de 1 cellule par bloc. La figure 5.10 nous montre un exemple d'extraction de HOG pour une image en présence d'un train. Cette représentation et l'extraction de HOG ont été obtenues grâce à la librairie VLFeat [Vedaldi & Fulkerson 2008]. Nous pouvons voir comment les différents *bins* de HOG permettent de distinguer la silhouette du train.

Cependant, pour notre descripteur, nous utiliserons la version classique de HOG, détaillée précédemment dans la section 4.3.3 avec les *bins* distribués comme dans l'équation 4.11. La taille du descripteur pour notre distribution suit la formule 5.1, où $N_{bins} = 10$.

$$Taille\ descripteur\ HOG = N_{blocs\ verticaux} \times N_{blocs\ horizontaux} \times N_{cellules\ par\ bloc} \times N_{bins} \quad (5.1)$$

Cependant, plus le nombre des cellules est grand, moins le descripteur est robuste aux déplacements verticaux et horizontaux et il est plus coûteux en temps de calcul et en mémoire. En particulier, si le train n'est pas centré dans la ROI à cause d'une mauvaise détection des rails ou un problème d'échelle (changement de pente), les blocs de HOG ne seront pas positionnés exactement dans la position souhaitée et le classifieur risque de donner un résultat négatif. Ceci nous oblige à trouver un compromis entre le nombre des blocs et l'exactitude du classifieur.



(a) Image originale. (b) HOG 15 lignes 10 colonnes. (c) HOG 8 lignes 5 colonnes. (d) HOG 4 lignes 3 colonnes.

FIGURE 5.10 – Descripteur de HOG.

La figure montre l'extraction du descripteur de HOG dans une image ROI en présence d'un train. Les orientations sont prises non-signées avec un total de 8 orientations par cellule. La figure (b) montre le résultat du descripteur obtenu avec une configuration de taille de bloc de 10 pixels. La figure (c) pour une taille de bloc de 20 pixels et finalement (d) pour 40 pixels. Nous pouvons voir que plus le bloc est grand, moins l'information apportée par le descripteur est détaillée. Plus petit est le bloc, plus grand sera le descripteur.

5.4 Résultats

Afin d'évaluer les performances de l'approche présentée dans ce chapitre, nous avons procédé à des expérimentations et évaluations exhaustives sur des images issues de la base de données qu'on a créée, provenant des vidéos acquises lors de nos expérimentations dans le terrain ainsi que de vidéos et d'images d'Internet.

Dans un premier temps, nous avons testé l'efficacité des différentes distributions du descripteur de HOG avec les images de la base de données. Pour le matérialiser, il est nécessaire de réaliser un apprentissage supervisé pour lequel nous n'avons sélectionné que 30% de la base de données pour l'entraînement et 70 % pour la phase de test.

Dans une seconde étape, nous allons tester le classifieur avec des images issues de la détection des rails. Comme indiqué, une fois les rails détectés, nous extrayons une image à la distance maximale, car si la voie n'est plus visible, elle peut être couverte par un train. Ces images ROI sont alors évaluées par notre détecteur pour déterminer la présence ou non d'un train.

Pour déterminer la qualité du descripteur et du classifieur, nous utilisons les mesures de précision et de sensibilité ou rappel (équations 5.2 et 5.3) :

$$Précision = \frac{VP}{VP + FP} \tag{5.2}$$

$$Sensibilité = Rappel = \frac{VP}{VP + FN} \tag{5.3}$$

Afin d'homogénéiser la base de données et d'avoir un descripteur de la même taille pour toutes les images, toutes ont été re-dimensionnées à une taille fixe de 150×100 pixels, en respectant le ratio hauteur/largeur du train.

Comme le descripteur choisi est un descripteur global, nous devons déterminer la distribution des blocs de HOG qui donne les meilleurs résultats en termes de classification. La figure 5.11 montre les résultats d'exactitude pour différentes configurations de HOG où le nombre des lignes et colonnes varie progressivement. Le tirage aléatoire de la base d'apprentissage a été reproduit 10 fois et les performances ont été moyennées. Nous avons utilisé un classifieur SVM à noyau linéaire.

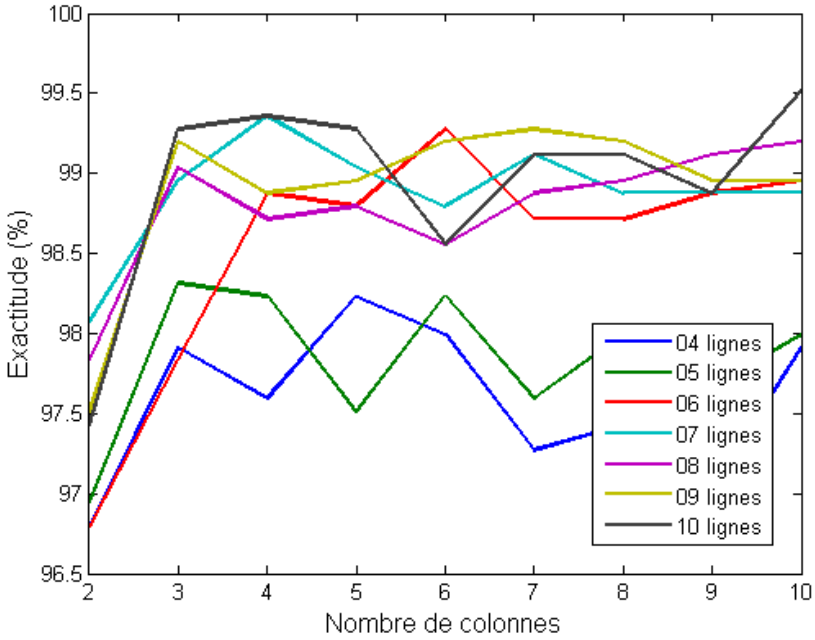


FIGURE 5.11 – Élection du nombre de lignes et de colonnes pour le descripteur de HOG.

Nous pouvons voir comment varie l'exactitude du classifieur SVM en fonction du nombre de colonnes et de lignes du descripteur HOG. La meilleure distribution en termes d'exactitude est 10 lignes et 10 colonnes. Cependant, le graphe montre que des descripteurs plus petits comme 7 lignes et 4 colonnes donnent des résultats similaires. La meilleure exactitude pour 7 lignes et 4 colonnes est de 98.98%

La figure 5.11 nous montre comment l'exactitude du classifieur augmente en même temps que le nombre des cellules augmente. Cependant, plus petit est le bloc, plus grand sera le descripteur et moins sensible sera celui-ci à des déplacements.

Un bon compromis peut-être de choisir comme meilleure configuration de HOG, 7 lignes et 4 colonnes, ce qui fait un bloc d'une taille de 20×25 pixels. Pour cette configuration le résultat de détection pour la phase de test est de 98.98% d'exactitude et de 99.36% de sensibilité.

Pour pouvoir mieux analyser les performances du descripteur et du classifieur, nous analysons la courbe précision/rappel exposée dans la figure 5.12. Cette courbe est obtenue en faisant varier le seuil de détection du SVM. Plus ce seuil est haut, meilleure sera la précision et pire le rappel. Cependant, le choix du point de fonctionnement dépendra de l'application. Nous pouvons voir que si nous exigeons du classifieur d'avoir un rappel de 100 %, c'est-à-dire, nous n'acceptons aucun faux négatif, la précision descend à seulement 97%.

Afin d'évaluer quantitativement les erreurs de détection de trains, nous pouvons exploiter la matrice de confusion du tableau 5.3. Seulement 4 images des trains n'ont pas été bien détectées, et 4 images ont été détectées comme faux positifs. Cependant, ces images correspondent à des vrai négatifs de la base de données qui correspondent à des exemples de trains. Ceci fait que leur silhouette ressemble beaucoup à la silhouette de notre métro VAL. Les exemples de ces images sont donnés dans les figures 5.13 et 5.14 respectivement.

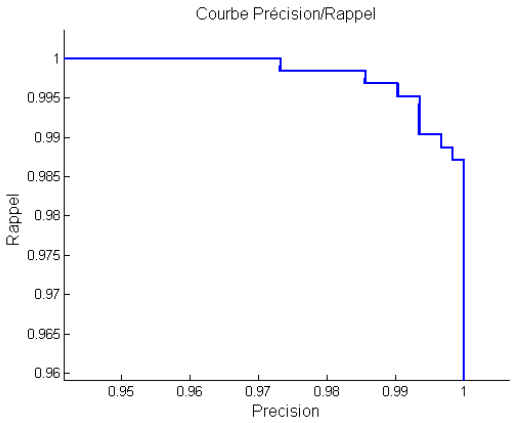


FIGURE 5.12 – Courbe Précision-Rappel obtenue pour SVM avec le descripteur HOG 7×4.

Tableau 5.3 Matrice de confusion pour la détection des trains

		Classifieur		
		Pas de train	Train	Rappel
Vérité terrain	Pas de train	623	4	99.36%
	Train	4	614	99.35%
	Précision	99.36%	99.35%	99.36%

FIGURE 5.13 – Images des faux négatifs.

FIGURE 5.14 – Images des faux positifs.

Il aurait été intéressant néanmoins de tester nos algorithmes sur des bases de données réelles dans des cas critiques. Malheureusement, nous ne disposons pas de telles bases de données et n'avons pas connaissance de l'existence de telles bases dans la communauté.

Finalement, nous présentons les résultats de 4101 images obtenues des vidéos de l'Orly VAL après l'extraction de la voie. Toutes sont des images sans train, car dans nos expérimentations, nous n'avons eu l'opportunité d'avoir un train en face dans la même voie. Le classifieur développé donne comme résultat pour ces images une exactitude de 100 %, c'est-à-dire, toutes les images ont été classifiées comme «non trains». Des exemples de ces images ROI extraites à partir de la détection de la voie sont présentés dans la figure 5.15.

FIGURE 5.15 – Images ROI obtenues à la distance maximale de détection de la voie.
Toutes ces images ont été classifiés correctement comme non-train.

Chapitre 6

Conclusions

Sommaire

6.1 Conclusions 111
6.2 Travaux futurs 113

6.1 Conclusions

Nous avons d'abord présenté un état de l'art sur les systèmes de détection d'obstacles et d'extraction de rails pour les trains. La plupart d'entre eux sont des systèmes multi-capteurs et utilisent des informations externes telles que la vitesse du train, la forme de la voie et les éléments qui la composent. À partir de cet état de l'art, nous avons proposé le développement d'un algorithme de détection des rails indépendant basé simplement sur l'analyse des gradients de l'image et avec un très faible seuil.

La première partie de notre étude s'est centrée sur l'étalonnage des paramètres extrinsèques de la caméra après son installation derrière le pare-brise du train, car peu de travaux ont été développés pour obtenir de manière simple ces paramètres dans le milieu ferroviaire. Nous avons donc développé un système complet d'étalonnage des caméras à partir de la morphologie de la voie en ligne droite. Ce système permettra de réduire des contraintes de maintenance des futures caméras installées dans des trains et ainsi de simplifier leur installation. Les résultats d'étalonnage obtenus avec la méthode proposée ont une précision suffisante pour les applications auxquelles est destinée la caméra.

Par la suite, nous avons défini un système générique et robuste pour l'extraction et détection des rails. Cette approche est basée sur la détection des gradients dans l'image à partir des simples contraintes géométriques spécifiques des caractéristiques de la voie ferrée. Nous analysons d'abord la position du point de fuite des rails pour déterminer une région d'intérêt à partir d'un masque adaptatif. Par la suite, l'image des gradients est analysée pour sélectionner des points ressemblant à des rails, puis nous les transformons en coordonnées 3D, où la recherche du modèle de rails est réalisée pour chaque rail à partir de l'algorithme de RANSAC. Ceci permet de réduire le temps de calcul, car sont seulement rétro-projetés les points ressemblant à des rails au lieu de construire une image complète en *bird-eye's view*. De plus, l'effet d'utiliser deux RANSAC indépendants pour chaque rail nous permet de réduire les erreurs dans la détection des rails quand l'hypothèse du monde plat n'est pas respectée.

Nous avons aussi proposé de calculer le déplacement du train à partir de la vidéo directement. Ce déplacement lié à la détection des rails permet de reconstruire le parcours suivi par le train. Même si les estimations de la voie à partir d'un polynôme d'ordre 2 ne donnent pas exactement les mêmes rayons de courbure instantanés, la reconstruction finale s'approche de la réalité. En plus, le fait d'avoir défini un modèle du train et de connaître son avancement dans l'espace permet de mieux filtrer temporellement les données pour réduire le nombre d'itérations de RANSAC.

Dans une troisième partie, nous avons proposé une méthode inédite de détection d'appareils de voie basée sur l'analyse de l'entrevoie à partir des images sans la déformation de perspective et

une extraction du descripteur de HOG. Le fait d'utiliser une image rectangulaire sans perspective simplifie la tache car la voie se présente toujours plus homogène. La technique basée dans le *template matching* a donné de moins bons résultats que le descripteur de HOG, principalement à cause des variations de texture de l'entrevoie.

Nous avons ensuite proposé d'étendre l'étude à la classification de l'appareil de voie en fonction de sa nature, car il peut être convergent ou divergent. Nous avons proposé d'utiliser la même image IPM sans déformation de perspective, et un descripteur de HOG pour mettre en valeur les orientations des voies divergentes ou convergentes. Plusieurs classificateurs ont été testés et finalement un classifieur SVM multi-classe avec la méthode *1-vs-1* a été retenu. Les résultats obtenus montrent qu'il est efficace de classifier les appareils de voie à partir du descripteur de HOG avec un rappel minimum de 97.92 % et une précision minimale de 98.10 %.

Dans un dernier temps, nous avons conçu une architecture complète pour la détection d'obstacles dans la voie courante. Cette méthode est basée sur l'extraction de la voie qui nous permet de déterminer une zone libre d'obstacles jusqu'à la distance maximale de détection. La connaissance de l'emplacement de la voie permet de ne pas réaliser un *sliding window* à différentes échelles pour déterminer la présence d'un train. De plus, moins d'images sont testées, plus faible est la probabilité de fausses détections, par exemple la détection d'un train dans une autre voie. Nous proposons donc d'évaluer l'hypothèse de la présence d'un train à la distance maximale de détection des rails en utilisant un classifieur binaire SVM et un descripteur de HOG. Un rappel supérieur à 99 % et une précision supérieure à 98 % indiquent de bonnes performances pour le classifieur avec les données de test de la base de données.

Pour le système complet, l'extraction de la voie suivie de la classification des images ROI extraites au-delà de la distance maximale de détection de rails, le classifieur testé sur des images sans trains, arrive à les classifier toutes correctement, sans aucune fausse alarme.

La figure 6.1 nous montre un schéma général du système complet, avec l'extraction de la voie, l'analyse d'une partie de l'entrevoie à une certaine distance à l'avant du train pour la détection des appareils de voie et une extraction de la ROI à la distance maximale de détection pour tester l'hypothèse de la présence d'un train.

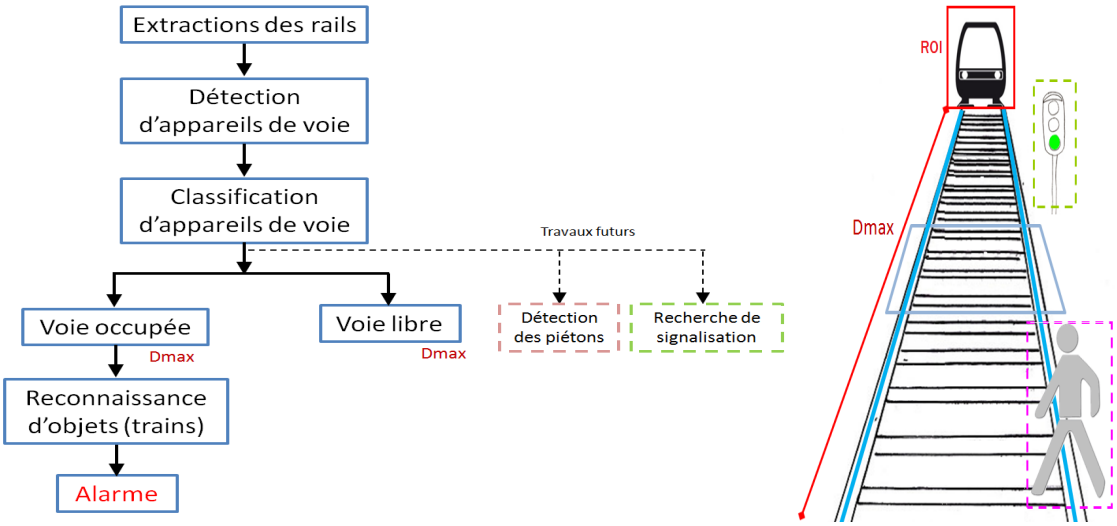


FIGURE 6.1 – Schéma des travaux réalisés et travaux futurs.

6.2 Travaux futurs

On peut envisager plusieurs perspectives pour les travaux présentés dans cette thèse, tant sur le plan industriel que sur le plan recherche.

Il serait intéressant d'étendre ces méthodes pour d'autres types de caméras comme les caméras de «Haute Range Dynamique» ou des «caméras de lumière structurée». Par exemple, on pourrait définir une méthode pour mieux évaluer la tolérance aux fausses détections des rails ou l'utilisation d'un filtre de Kalman pour obtenir des résultats plus fins et précis dans les grandes distances.

Aussi, il serait envisageable de développer rapidement avec la même technique d'extraction des rails la recherche de la voie secondaire une fois que le train s'approche d'un appareil de voie divergent. Puisque le système ne connait pas à l'avance la direction à suivre, l'objectif est de déterminer aussi une zone libre d'obstacles dans les deux voies.

De plus, il serait intéressant d'étendre la recherche des obstacles à la détection des piétons car, actuellement, les derniers travaux de détection de piétons montrent des résultats prometteurs.

Du côté industriel, la détection et reconnaissance des appareils de voie pourrait s'utiliser pour une localisation visuelle sans avoir à ajouter des amers visuels secondaires. L'utilisation d'une carte virtuelle de la voie avec le positionnement des appareils de voie permettrait de déterminer la position du train de façon précise dans différents tronçons. Cette fonction pourrait s'utiliser sous certaines conditions de sécurité ferroviaire pour, par exemple, envoyer les trains sur les voies de garage ou dans l'atelier en mode automatique, quand il n'est pas possible d'utiliser certains systèmes de positionnement comme le GPS ou pour réduire le nombre des balises au sol.

Également, la détection de trains pourrait s'utiliser pour déterminer la distance au train précédent pour des manœuvres d'accostage ou pour approcher au maximum les trains sur les voies de garage. Ainsi, les feux de signalisation peuvent être détectés.

Enfin, il serait utile de tester nos travaux sur d'autres bases de données plus étendues et sous différentes conditions météorologiques, ou avec d'autres types de parcours comme des tunnels ou voies de garage.

Annexe

Publications

Sommaire

A.1 Article ITSC 2012 116
A.2 Article ICIRT 2013 121
A.3 Article ITSC 2013 127

Conférences internationales avec comité de relecture

1. ***Rail extraction technique using gradient information and a priori shape model***, Jorge Corsino Espino, Bogdan Stanciulescu, ITSC 2012 *15th International IEEE Conference on Intelligent Transportation Systems*
2. ***Rail and Turnout detection using gradient information and template matching***, Jorge Corsino Espino, Bogdan Stanciulescu, Philippe Forin, ICIRT 2013, *1st International IEEE Conference on Intelligent Rail Transportation*
3. ***Turnout detection and classification using modified HOG and template matching***, Jorge Corsino Espino, Bogdan Stanciulescu, ITSC 2013 *16th International IEEE Conference on Intelligent Transportation Systems*
4. ***Safe Navigating System for Indoor Environments***, Jorge Corsino Espino, Bruno Steux, Oussama El Hamzaoui, ICARA 2011 *5th International Conference on Automation, Robotics and Applications*
5. ***Autonomous Navigation and Mapping with CoreSLAM***,Oussama El Hamzaoui, Jorge Corsino Espino, Bruno Steux, *Recent Advances in Robotics and Automation*,pages (91-101), *Springer Berlin Heidelberg*

A.1 Article ITSC 2012

A comparative study of rail extraction techniques

Jorge Corsino Espino[1] and Bogdan Stanciulescu[2]

Abstract— **This paper presents a comparative study of different rail detection techniques as well as a new method based on an efficient algorithm without any empirical thresholds. The main problem with rail detection is that both the track-bed and and the exterior conditions (weather/light conditions) vary along the path. On the other hand, there are properties that can be exploited to improve the conventional lane detection. We present an edge detection based on the estimated position of the rails that follows the rail edges upwards in the image, determining a free-from-obstacles space. The existing techniques are also analyzed and compared.**

I. Introduction

Trains are considered one of the safest means of transportation. Unfortunately, accidents still often occur due to the fatigue of the driver, the bad weather conditions or other factors. The safety can be improved using intelligent systems that provide additional information about the exact location of the train, its speed and the upcoming obstacles. This can be accomplished by installing sensors along the tracks, which can be expensive at a large scale. An alternative is to install embarked cameras at the front of the train to support the driver. There are many constraints imposed on such systems, such as the cost and resource efficiency as well as the real-time performance.

Rail extraction is one of the fundamental tasks in an on-board computer vision system for trains. It can be useful for the positioning and the speed estimation of the trains as well as the obstacle detection. However it faces many challenges such as the bad illumination and poor visibility in tunnels or under bad weather conditions.

Our goal is to automatically find the rails in order to determinate the free path in front of the train using only image processing.

A. The environment

The rubber-tyred metro is a rapid transportation system that uses a mix of road and rail technology. The vehicle has wheels which run on rolling pads inside guide bars for traction, as well as the traditional railway steel wheels with deep flanges on steel tracks for guidance. Further, the pads are also often used for communication or control structures such as duct cables and drains.

The environment is particularly challenging in train systems since there are several parallel rails, a richly textured track-bed as well as frequent variations in illumination, rail equipment and tunnels as shown in figures 1, 2 and 3.

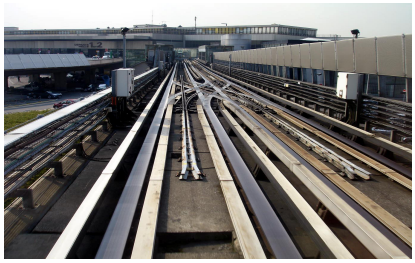


Fig. 1. Driver's view with a fixed camera placed behind the windshield.

The complexity of the track-bed makes the rail detection difficult due to the large quantity of straight segments and the shadows of the guidance tracks produced on the pads. This task is more complex at night or on cloudy days because of the variation of the color of the rails. Further, detecting dark rails over dark surfaces is harder than detecting bright lines on a dark road as in the road lane detection systems.

B. Previous works

Some studies [1], [2] show that the lane marking detection techniques, used in intelligent vehicles, can also be applied to the rail detection in certain applications. Both the lane-markings and the rail tracks are represented as a group of lines. Furthermore, the curvatures in both applications tend to have a large radius, barely changing over small distances. However, the textured structure of the track-bed, due to ballast, grass and sleepers, as opposed to the uniform asphalt background of the lane-markings, makes the rails more difficult to detect. Furthermore, the reflective nature of the steel causes larger variations due to shadows and changes in illumination.

Previous works have exploited the aforementioned characteristics of the railway tracks to detect them. In [3], the line segments, extracted using the edge detection, are matched to a set of track models. A probability measure designates the corresponding match using the Chamfer similarity metric. However, the edge detection is sensitive to the fine tuning of the parameters. Nassu et al. apply a hill-climbing approach for the parameter selection. The feasibility and efficiency of this approach is also limited by the choice and the number of rail segment models.

In [4], the lines representing the tracks are also extracted. This is performed using quadratic programming. The initial

[1] J. Corsino Espino with SIEMENS SAS Infrastructure & Cities. 150, Avenue de la République, 92323 Chatillon cedex, FRANCE jorge.corsino-espino at siemens.com

[2] B. Stanciulescu with Mines ParisTech, Center of Robotics (CAOR). 60, boulevard Saint Michel, 75272 Paris cedex 06, FRANCE. bogdan.stanciulescu at mines-paristech.com

Paper 16th April

segments at the bottom of the image are extended by adding neighboring candidate segments which fulfill several constraints, such as the pixel value distribution, the angles of the gradients and the distance between the tracks. The empirical choice of parameters renders the system less generic and more sensitive to variations.

Kaleli et al. [5] also perform a preliminary edge detection to extract track candidates. They then use the Hough transform to merge the line segments into longer lines. Further, they use a dynamic programming approach to simultaneously detect the right and left tracks by ensuring the predefined width between them.

In the anti-collision system presented in [6], the image is warped into a birds eye view to obtain parallel tracks, as opposed to the detection of the rail crossings in the vanishing point at the horizon. The parameter selection is initialized with the help of a human operator that designates the search region of the rails. The lines are extracted using a Hough transform.

In [7], the switches at the railway crossings are detected. The eigenvalues of the gradients computed over image patches are used to identify homogeneous and textured regions. The former are discarded, while the latter are filtered according to the gradient angles. Only a restricted area of the image is examined to reduce the computational overhead. The calculation of the covariance and the eigenvalues can be expensive at a large scale.

In this paper, we propose a rail detection method for rubber-tyred trains, by iteratively selecting local maxima in the gradient image using a sliding window approach. We test our method on challenging sequences with several parallel rails, richly textured background and infrastructure as well as illumination variations due to shadows and tunnels. The system computes the length of the free obstacle space in front of the train.

The remainder of the paper is organized as follows: in Section 2, we give an overview of some problems for existing techniques. In Section 3, we describe the proposed system; in Section 4, we present some experimental results. Finally, in Section 5, we summarize the conclusions and give an outlook on future work.

II. Comparison of existing techniques

The most common techniques used for road lane and rail detection are edges extraction and lines detection in the images. In this paper, we analyze both these techniques and compared the results obtained.

A. Edge detection

Edge detection is an effective image processing tool that extracts image information and characteristics. Before applying the edge detection, it is important to filter the image to smooth out any noise arising from the image capturing. This noise can lead to the detection of false edges by the detector [8]. Futhermore, the lighting conditions vary along the track surface and the shadows caused by trees or tunnels. The quality of the resulting edges depend on the thresholds used.

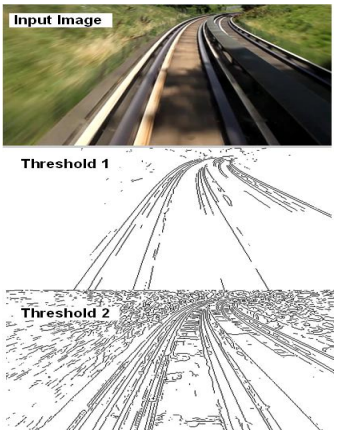


Fig. 2. Different results using Canny algorithm and varying the thresholds.

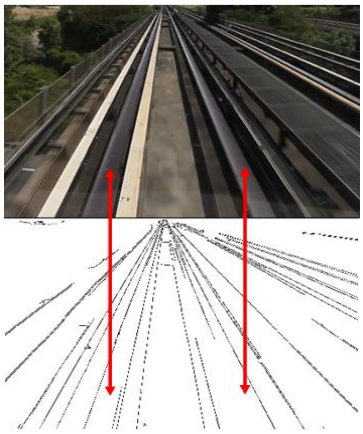


Fig. 3. Mismatching of a rail. The shadow produces a hard edge and the right rail is not well detected.

In some cases, a simple shadow on the image can produce the mismatching of a rail, as shown in figure 3.

When using the Canny edge detector [8] with a low threshold, we can generate a maximum amount of edges. To reduce the computational overhead, only the gradients with an angle in a range are kept. This range is determined by the perspective projection of a right track and a curved track at its minimun radius at different distances [9]. These filtred edges are kept as possible track candidates. As seen in figure 4, the number of edges are reduced to only a few (highlighted in red). This technique filters a large amount of edges. Nevertheless, the angular conditions are strict and sometimes the relative angle of the gradient in the pixel grid does not match the angle of the line in the image.

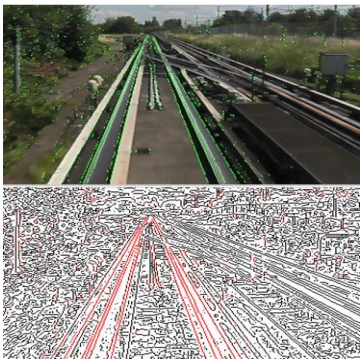


Fig. 4. Canny filter with low threshold and filtered edges (in red) using the angle gradients.

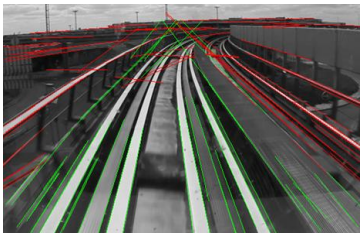


Fig. 5. Lanes detected with Hough Lines Transform.

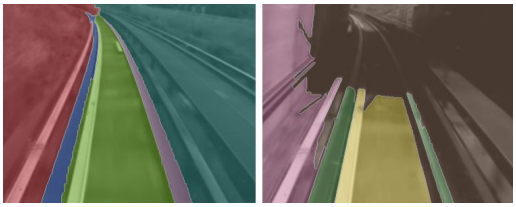


Fig. 6. Watershed segmentation. Left: an accurate rail track segmentation. Right: an example showing the influence of a tunnel entrance on the segmentation result.

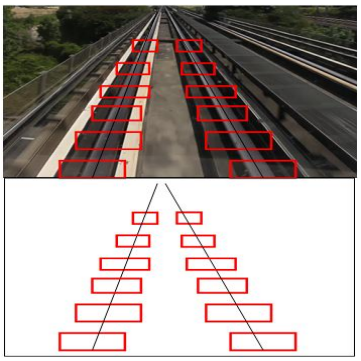


Fig. 7. Sliding window from bottom to up.

B. Hough Transform

Choilane et al. [10] use the Hough Transform combined with a Kalman filter to detect lines on the road. The model of the road in the Hough space and the vanishing point extraction allow for the detection of the road lines with a good performance and on challenging sequences. For the rubber-tyred rail detection in this paper, the Hough line detector can be used to find the vanishing point with a good performance. Detecting the shape of the rails is less efficient as a result of the many lines obtained. The need of thresholds on the Hough line transform does not always produce the best results. Further, the filtering of all the lines is computionally inefficient. Figure 5 shows the Hough line transform with all dectected lines.

C. Watershed

Good results in rail track segmentation are obtained using the Watershed technique [11]. However, due to aspects such as the high computational requirements as well as the presence of shadows or dark tunnels, the track segmentation is interrupted as shown in figure 6.

III. Proposed approach and experimental results

Many of the existing approaches rely on empirically defined thresholds and fine-tuned parameters. To develop a more generic approach, we propose to use a sliding window and a local maxima selection to make the rail detection more robust to changes in illumination, shadows and occlusions.

Our algorithm is organised as follows: firstly, the gradient image is computed inside the sliding windows. Then we search for maxima and minima in the gradient image to find edges. From these edges, rails are extracted and compared with an expected position. We continue moving the sliding window over the image upwards. (See figure 7 for an overview of the movement of the sliding window).

In the first step, a sliding window with a decreasing width is passed vertically over the image, starting at the bottom. The starting position of the rails at the bottom of the image is assumed to be constant and known, since the embarked camera is fixed and the width of the rails is constant. For each row in the sliding window, the edges are analyzed to find the correspondence with the predicted shape at this position.

This local analysis allows us to detect low contrast edges. The next sliding window is centred with respect to the new detected rail position.

To eliminate false alarms and avoid the deviation of the iterative search process, each new point detected as a track is compared to the previous points in the sliding window. The only points kept are those that minimize the mean square error with respect to the approximating line that represents them are kept. To make sure that the retained points are correct, they have to correspond with the projective straight line of the attended rail.

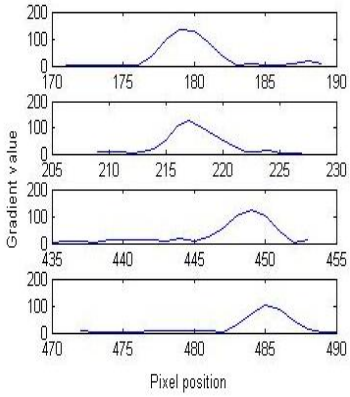


Fig. 8. Gradients inside the sliding windows for each side of the track. The window is centered with respect to where the rail edge is detected. The difference between its position and the maximum gradient detected, allows for the tracking of the position of the rails over the image

This straight line has a decreasing length, larger at the bottom and smaller in far distances, allowing to approximate a curve to segments of straight lines depending on the distance from the camera.

The estimated position is obtained using the perspective transformation between the real world and the 2D image [9]. For this, a calibrated camera and the knowledge of the expected shape is needed. Because the radius of the curvature is limited, we can always assume having a straight line in front of the train to initiate the algorithm. This straight line is then adapted to follow the track vertically across the image.

The objective of this straight line is to determine the evolution of the rails over the image. It is calculated using the retained edges as rails and determines the next position of the slidings window in the rails over the image.

This exploration is performed for the four edges of the rails. It allows us to determine the correct position of the rails over the image. Figure 8 shows the edges in the four sliding windows for each rail side. The center of the window is the predicted position of the rail and the maximum gradients are the edges of the rails. With this dynamic progression, we can overcome segments of rails with dark shadows where our local maxima selection does not detect any edge.

The search process is stopped when the the vanishing point is reached or when the rails are not well detected, due to an obstacle covering them.

Our sequences consist of videos taken 30 frames per second recorded using a camera with 5 mm focal length and 1/3 inch sensor size fixed behind the windshield at 2.78 m from the ground. The image size is 480 by 640 pixels. Figures 9, 10 and 11 illustrate the extraction of straight tracks from the bottom to the top of the image for curved rails.

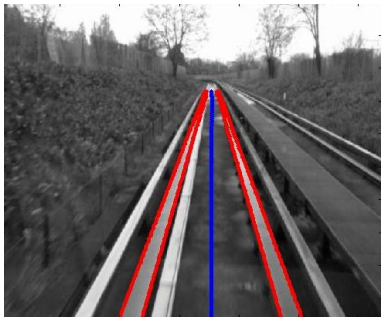


Fig. 9. Extracted rails

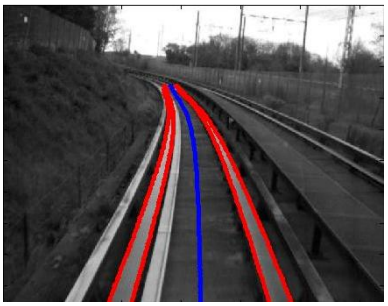


Fig. 10. Extracted rails in curve

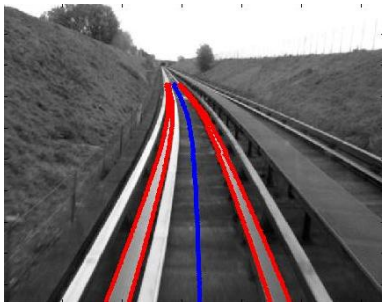


Fig. 11. Extracted rails in curve

IV. Conclusion and Future works

In this paper, a comparison of various techniques for rail extraction are presented and a new generic and robust approach is introduced. Our algorithm uses edge detection and applies simple geometric constraints provided by the rail characteristics. This allows to reduce both the cost in the computational load, as well the error in the detection of the rails.

Future works will include a learning approach for the pattern recognition in the area around the rails to improve the detection performance in presence of unusual objects i.e.

obstacles on the trackbed.

We plan on publishing our dataset shortly, to enable a comparison of new and existing approaches on a public benchmark, since no such dataset is currently available.

REFERENCES

[1] J. McCall and M. Trivedi, "Video-based lane estimation and tracking for driver assistance: survey, system, and evaluation," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 7, no. 1, pp. 20 –37, march 2006.

[2] V. Kastrinaki, M. Zervakis, and K. Kalaitzakis, "A survey of video processing techniques for traffic applications," *Image and Vision Computing*, vol. 21, pp. 359–381, 2003.

[3] B. Nassu and M. Ukai, "Rail extraction for driver support in railways," in *Intelligent Vehicles Symposium (IV), 2011 IEEE*, june 2011, pp. 83 –88.

[4] F. Maire, "Vision based anti-collision system for rail track maintenance vehicles," in *Advanced Video and Signal Based Surveillance, 2007. AVSS 2007. IEEE Conference on*, sept. 2007, pp. 170 –175.

[5] F. Kaleli and Y. Akgul, "Vision-based railroad track extraction using dynamic programming," in *Intelligent Transportation Systems, 2009. ITSC '09. 12th International IEEE Conference on*, oct. 2009, pp. 1 –6.

[6] F. Maire and A. Bigdeli, "Obstacle-free range determination for rail track maintenance vehicles," in *Control Automation Robotics Vision (ICARCV), 2010 11th International Conference on*, dec. 2010, pp. 2172 –2178.

[7] J. Wohlfeil, "Vision based rail track and switch recognition for self-localization of trains in a rail network," in *Intelligent Vehicles Symposium (IV), 2011 IEEE*, june 2011, pp. 1025 –1030.

[8] J. Canny, "A computational approach to edge detection," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. PAMI-8, no. 6, pp. 679 –698, nov. 1986.

[9] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, ISBN: 0521540518, 2004.

[10] K. Choi, K. Min, S. Lee, W. Park, Y. Seo, and Y. Hong, "Lane tracking in hough space using kalman filter."

[11] S. Beucher and M. Bilodeau, "Road segmentation and obstacle detection by a fast watershed transformation," in *Intelligent Vehicles' 94 Symposium, Proceedings of the*. IEEE, 1994, pp. 296–301.

A.2 Article ICIRT 2013

Rail and turnout detection using gradient information and template matching

Jorge Corsino Espino
SIEMENS S.A.S.
SWE RC-FR IC-MOL RA R&D
92320 Chatillon, France
jorge.corsino-espino@siemens.com

Bogdan Stanciulescu
Robotic Center
Mines ParisTech
75272 Paris, France
bogdan.stanciulescu@mines-paristech.com

Philippe Forin
SIEMENS S.A.S.
SWE RC-FR IC-MOL RA R&D
92320 Chatillon, France
philippe.forin@siemens.com

***Abstract*—This paper presents a railway track and turnout detection algorithm which is not based on an empirical threshold. The railway track extraction is based on an edge detection using the width of the rolling pads. This edge detection scheme is then used as an input to the RANSAC algorithm to determine the model of the rails. The turnout detection scheme is based on the Histogram of Oriented Gradient (HOG) and Template Matching (TM). The results show (i) reliable performance for our railway track extraction scheme and (ii) a correction rate of 97.31 percent for the turnout detection scheme using a Support Vector Machine (SVM) classifier.**

***Keywords*—Rails detection, Turnout detection**

I. Introduction

Trains are considered one of the safest means of transportation. Unfortunately, accidents still often occur due to driver's tiredness, bad weather conditions or other factors. Safety can be improved using intelligent systems that provide additional information about the exact location of the train, its speed and upcoming obstacles. This can be accomplished by installing sensors along the tracks but it can be expensive at a large scale. A more cost efficient approach is to install on-board cameras at the front of the train to support the driver. There are many constraints imposed on systems improving safety such as cost, resource efficiency as well as real-time performance.

Rail recognition and extraction is one of the fundamental tasks in an on-board computer vision system for trains. It can be useful for trains positioning and speed estimation as well as obstacle detection. However, it faces many challenges such as bad illumination and poor visibility in tunnels or under bad weather conditions. Railway turnouts are characterized by complex rails extraction scenarios. For this reason, this paper proposes an approach for rails and turnout detection using only image processing to determine a free path in front of the train.

The remainder of the paper is organized as follows: in Section 2 we describe the state of the art; in Section 3, we give an overview of the proposed system; in Section 4, we detail the proposed system for rails extraction; in Section 5, we present the approach for turnout detection; in Section 6 we present some experimental results. Finally, in Section 7, we summarize the conclusions and give an outlook on future works.

II. Previous work

Some studies [1], [2] show that the lane marking detection techniques, used in intelligent vehicles, can also be applied to the rail detection in certain applications. Both the lane-markings and the rail tracks are represented as a group of lines. Furthermore, the curvatures in both applications tend to have a large radius, barely changing over small distances. However, the textured structure of the track-bed, due to ballast, grass and sleepers, as opposed to the uniform asphalt background of the lane-markings, makes the rails more difficult to detect. Furthermore, the reflective nature of the steel causes larger variations due to shadows and changes in illumination.

Previous works have exploited the aforementioned characteristics of the railway tracks to detect them. In [3], the line segments, extracted using the edge detection, are matched to a set of track models. A probability measure designates the corresponding match using the Chamfer similarity metric. However, the edge detection is sensitive to the fine tuning of the parameters. Nassu et al. apply a hill-climbing approach for the parameter selection. The feasibility and efficiency of this approach is also limited by the choice and the number of rail segment models.

In [4], the lines representing the tracks are also extracted. This is performed using quadratic programming. The initial segments at the bottom of the image are extended by adding neighbouring candidate segments which fulfil several constraints. The empirical choice of parameters renders the system less generic and more sensitive to variations.

Kaleli et al. [5] also perform a preliminary edge detection to extract track candidates. They then use the Hough transform to merge the line segments into longer lines. Further, they use a dynamic programming approach to simultaneously detect the right and left tracks by ensuring the predefined width between them.

In the anti-collision system presented in [6], the image is warped into a bird's eye view to obtain parallel tracks, as opposed to the detection of the rail's crossings in the vanishing point at the horizon. The parameter selection is initialized with the help of a human operator that designates the search region of the rails. The lines are extracted using a Hough transform.

In [7], the switches at the railway crossings are detected. The eigenvalues of the gradients computed over image patches are used to identify homogeneous and textured regions. The

former is discarded, while the latter is filtered according to the gradient angles. Only a restricted area of the image is examined to reduce the computational overhead. The calculation of the covariance and the eigenvalues can be expensive at a large scale.

In this paper, we propose a rail detection method for rubber-tyred trains, using the RANdom SAmple Consensus (RANSAC) [8] algorithm iteratively, selecting the best model from frame to frame. We test our method on challenging sequences with several parallel rails, richly textured background as well as illumination variations due to shadows and tunnels. At the same time, the system detects turnout sections over the rails to change the approach of RANSAC. For that the system analyzes the area between rails, searching for the presence of a guidance rail in turnouts.

III. Overview of the method

A. The environment

Rubber-tyred metro is a rapid transportation system that uses a mix of road and rail technology. The vehicle has wheels which run on rolling pads inside guide bars for traction, as well as the traditional railway steel wheels with deep flanges on steel tracks for guidance. Further, the pads are also often used for communication or control structures such as duct cables and drains.

The environment is particularly challenging in train systems since there are several parallel rails, a richly textured track-bed, rail's equipment and tunnels as shown in Fig. 1.

The complexity of track-bed makes rail detection difficult due to the large quantity of straight segments and the shadows of the guidance tracks produced on the pads. As seen in Fig. 1, the rolling pads and other structures are close to each other. This task is more complex at night or on cloudy days because of rails' color variation. Further, detecting dark rails over dark surfaces is harder than detecting bright lines on a dark road as in the road lane detection systems.

B. Rails Geometry

Rails geometry describes tracks' layouts with specific alignments, curvature warp, rail gauge etc. In our case the principal characteristics are:

- Minimum horizontal curve radius: $12m$
- Minimum vertical curve radius: $200m$
- Maximum slope: 12%
- Width of rolling pads $w = 250mm$
- Standard rails gauge $g = 1650mm$

Movement tolerance of the rail vehicle

- Banking: from 0% to 13% $(0,55\%/m)$
- Roll: $< 2\,^{\circ}/s$
- Pitch: $< 2\,^{\circ}/s$
- Yaw:$+/-3\,^{\circ}/s$

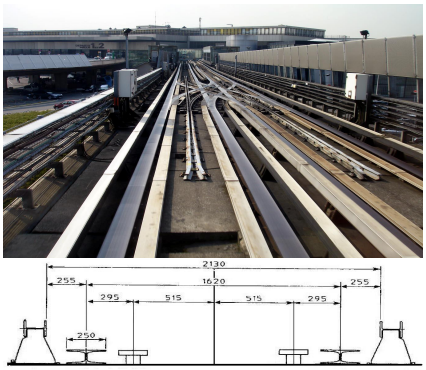


Fig. 1. Driver's view with a fixed camera placed behind the wind-shield and vertical section of track bed and its distances.

Turnouts are used to divide or merge tracks into two or more tracks. The purpose of crossings is to allow tracks to intersect at the same level. In our case, turnouts are equipped with a central switch to guide the train over the turnout. This switch is like a central rail of guidance. An example of our turnout is shown in Fig. 1.

Assuming that the track-bed is planar, we can compute an inverse rectifying homography that maps the center camera image to a bird's eye" point of view. Formally, we chose a train-centric coordinate system as follows: $+Z$ is forward, $+X$ is right and $+Y$ is up. The X position of the center of the rails is modeled as a quadratic function in the ground plane such as: $X(z) = az^2 + bz + c$. Given the width of the rails (w), the standard gauge (g) and the inverse rectifying homography (H_r) for the trackbed where $Y = 0$, we can immediately derive the location X of the left and right rails' edges for any Z.

C. Camera

A forward-facing camera was installed behind the wind shield at a height h of $2.78m$ above the trackbed surface and with a tilt angle of $\theta = 10\,^{\circ}$. Our sequences consist of videos taken at $30fps$ recorded by a camera with $5mm$ focal length and $1/3inch$ sensor size. The image resolution is 480 by 640 pixels.

D. Rails and turnout detection approach

Although the goal of this work is turnout detection, it is crucial to first detect the rails. It starts by detecting the center of the rails at the bottom of the image. Then the image's gradients are analyzed to find the rails candidates. These points $p(u, v)$ are then converted with the Homography (H_r) to 3D points $P(X, Y, Z)$. It is in this space where we fit the best model of the rails. For this purpose we propose to use RANSAC. In order to reduce the number of edges and keeping only the rails, other operations are exploited: masking, binarization, gradient angle analysis and filtering. Our algorithm is organized as follows:

1) Center of rails detection
2) Masking

3) Gradient analysis
4) Homography
5) RANSAC
   a) Left RANSAC
   b) Right RANSAC
   c) Central parabola
6) Vanishing point of rails

IV. Proposed approach for Rails Detection

A. Edged-based rails detection

The most common techniques used for road lane and rail detection are edge extraction and lines detection in the images. Many of the existing approaches rely on empirically defined thresholds and fine-tuned parameters. We propose to develop a more generic approach to find rails' edges, searching for local maxima and minima gradient separated by a width of w. Before applying the edge detection, it is important to filter the image to smooth out any noise arising from the image capturing. This noise can lead to the detection of false edges.

We can generate a maximum amount of edges as rails candidates, but to reduce the computational overhead, only almost parallel gradients, (having similar angle orientation) are kept. Therefore, horizontal and vertical gradients are analyzed. We generate a binary image with the center position of edges found, considering them as rails candidates. This image is then filtered by a neighbouring mask of size 4x4. As seen in Fig. 2, the number of edges is reduced to only a few, filtering a large amount of erroneous edges and simplify the next steps.

This technique makes rail detection more robust to changes in illumination, shadows and occlusions because there is no threshold applied to gradient magnitude.

B. Masking

In order to speed up the algorithm and to reduce the number of edges we apply a mask over the image. This mask defines a region of interest with the track's shape but with a bigger width. To place appropriately this mask, it is necessary to know the rails' position. For that, the vanishing point (VP) of the rails is calculated and used as the end point of our mask. The VP of the rails is obtained by analyzing the tangent to detected rails at its maximum curve. As reminder, the tangent of a function is given by the equation (1)

$$T(x) = f'(a)(x - a) + f(a) \tag{1}$$

An example of the mask can be seen in Fig. 3. For the first frames, the mask is not applied until we have determined a good rails detection.

C. Fitting the rails

From the precedent binary image, we obtain the retro-projection points $P(X, Y, Z)$ for every rails candidate and store them in vector M to extract the rails in the rolling plane. For this purpose we decided to apply an iterative procedure to reduce error detection probability. We firstly use RANSAC to detect left and right rail independently. RANSAC is a robust estimation technique based on the principle of hypotheses, generation and verification, fitting models in the presence of many data outliers. For each search, we impose some conditions to the RANSAC algorithm as stated below:

1) Randomly select three points at different distances Z reducing the probability of a wrong model and the number of iterations.
2) Estimate the model m fitting these three initial inliers.
3) Check model: Verify that the radius of curvature corresponds to the rails' layout.
4) Find the number of M's data items that fit the model with a tolerance t, avoiding to accept other track-pad

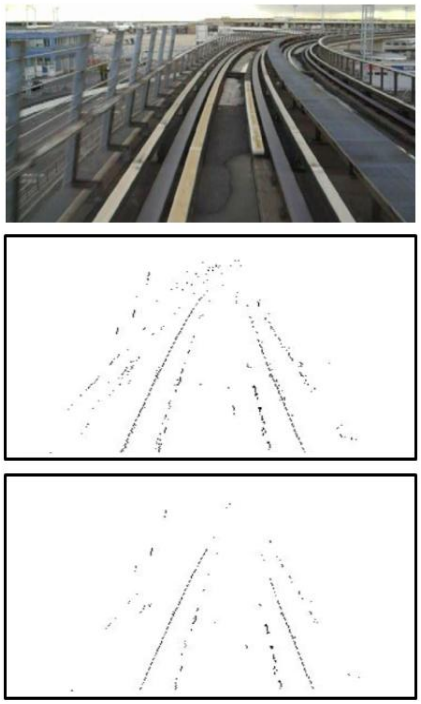


Fig. 2. Process of extracting edges rails candidates.First image is the input image. Second one shows the rails candidates obtained with the gradient analysis. Third image is the filtred image by neighbouring 4x4.

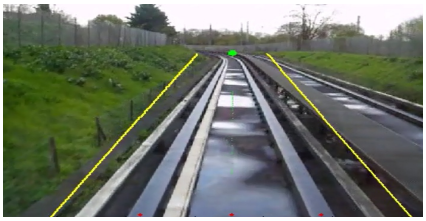


Fig. 3. Example of mask.

structure as a rail.

5) Calculate the model's error considering its rail's partner. As the rail's gauge is constant, we can calculate the model's error found at a distance g.
6) Repeat 1-5 L times

To calculate the model's error, we consider the opposite track. Using two Gaussians separated by a width gauge (g), and centered over the rails all along Z distances, we calculate the error of all M's points to the Gaussians. The best fitted model will have the minimum error.

Considering the presence of vertical curves ($slopes$) and vibrations of the rail's vehicle, the homography (H_r) can produce errors in converting $p(u, v)$ to $P(X, Y, Z)$. This is why we decided to analyze each rail separately to obtain the rails' central function curve. From each RANSAC, we keep the inliers and then we translate them to the rails' center. Once the translation is done, a new model is obtained by weighted least squares (equation 2). Weights depend on distance Z. Using this translation of both inliers to the rails' center, the algorithm can lead to small changes in the flat world assumption. The process of fitting the model is shown in Fig. 4.

For the next frame, as the position of rails is known, we can pre filtrate the new M vector to reduce the RANSAC process. This rail extraction method is efficient and robust. An example of edge rails tracker working accurately is shown in Fig. 9.

$$WSS = \sum_{i=1}^{n} W_i r_i^2 \quad , Wi = \frac{1}{D_{z_i}}, \quad r_i^2 = y_f(x_i, \beta) \quad (2)$$

V. Proposed approach for Turnout Detection

The use of bird-eye view makes the detection of turnouts easier because tracks appear parallel, have constant width and are mainly vertical. Besides, every pixel appears in world coordinates. In presence of a turnout, there is a central rail to guide the train over it.

The purpose of turnout detection is to determine their position before reaching them. Once rails are detected, we get the bird-eye view of rails at a distance Z between $10m$ and $12m$ with an offset of $1m$, obtaining an unwrap image of $2x2m$. Some examples of rails bird-eye view at $10m$ are shown in Fig. 5. We consider to analyze the center of the unwrap image, because in all turnouts there is a central rail. Hence we propose two techniques to find this rail: Template image matching and Histogram of oriented gradients (HOG).

A. Template matching

Template image matching [9] consist of finding a small part of an image which matches a template image. A basic problem of template matching is to determine the position of a given pattern in an image. In our case, the position of the center rail in the bird-eye view image is almost constant. Another problem is to determine how much the image and template's brightness can vary due to lighting and exposure conditions. Therefore images have to be normalized first. This can be done at every step by subtracting the mean and dividing by the standard deviation.

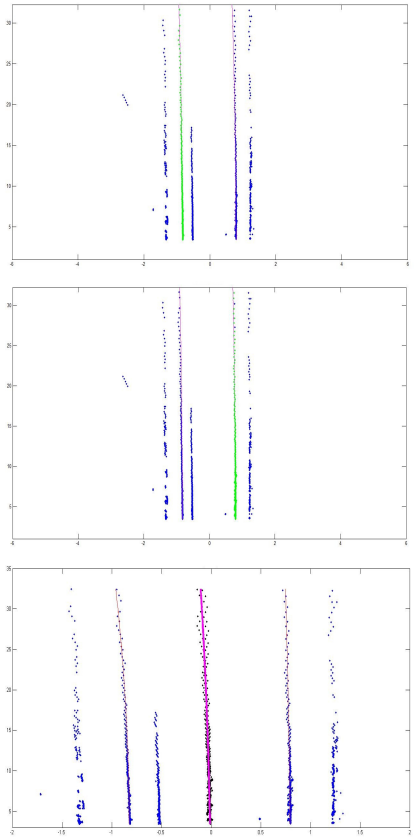


Fig. 4. Retro-projection of edges to fit models with RANSAC. First image shows the result of RANSAC for the left rail. Second is the result for right rail. Third image shows the translation of both inliers

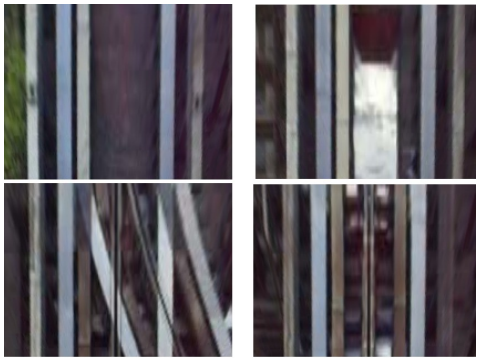


Fig. 5. Bird-eye view images of rails at $10m$. On the top, bird-eye image of simple rails. At the bottom, with turnouts.

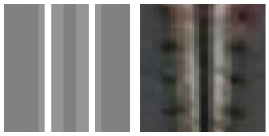


Fig. 6. At left the template image for center rail detection. At right an example of a center rail in bird-eye view

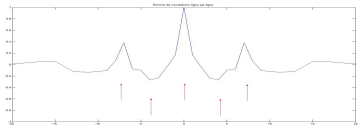


Fig. 7. Auto cross correlation of the template.

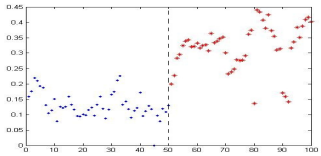


Fig. 8. Maximum value of correlation peak for 50 images without turnout and 50 with turnout.

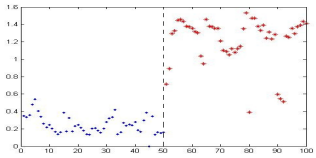


Fig. 9. Correlation feature extracted for 50 images without turnout and 50 with turnout.

Given the center rail's template t as shown in Fig. 6, the idea of the algorithm is to calculate normalized cross-correlations of template and sub-image line per line. Then correlation is calculated as a sum of whole linears correlations. Figure 7 shows the result of the template's auto-cross correlation. This result is characterized by 3 peaks and 2 valleys.

Taking as correlation feature the value of the center peak (cross-correlation coefficient), we can distinguish between images with and without turnout. But if we take the sum of these 3 peaks and the 2 valleys the distance between two classes is bigger, as shown in figures 8 and 9. Hence, we use this sum as correlation feature to determine the presence of a central rail (turnout) or not.

$$Corr_feature = \sum_{i=1}^{3} peak + \sum_{i=1}^{2} valley \qquad (3)$$

		Diagnosis		
		Positive	Negative	Total
Test	Positive	526	32	568
	Negative	110	4603	4713
	Total	636	4635	5271

TABLE I.

B. HOG

Histogram of Oriented Gradients [10] is a feature descriptor that gives local shape information often well described by the distribution of intensity gradients or edge directions. The technique counts occurrences of gradient orientation in localized portions of an image. Each pixel votes for an orientation according to the closest bin in the range. Our bins range is defined from 0 do 180 with 5 bins $[15; 80; 100; 135; 180]$. In the voting rule, we do not use the gradient's amplitude to avoid noise in illumination changes. We use just one cell for HOG algorithm because the bird-eye view has a reduced size and we only analyze the area enclosed by the rails.

As expected, in presence of a turnout, HOG reveals vertical gradient of center rail and the spatial distribution of bins is centered in the 3rd bin. Otherwise, all bins have a similar amplitude.

VI. Results

Our algorithm was tested using different videos of the metro line OrlyVal in a cloudy day. In these videos the train rolls over 6 different turnouts. As explained, it is necessary to extract rails before detecting turnouts. The mean of rails' maximum distance detected is 25 meters. This distance is limited by the low resolution of the camera. Fitting results under different circumstances are shown in Fig. 9. Blue color presents the fitted model for each rail and green color shows the central rails curve. All along the videos, rails extraction is done accurately.

For turnout detection, we use as a classifier a Support Vector Machine (SVM) trained with less than $1/3$ of data base images. The SVM input is the correlation feature and 3 bins of HOG (15-80; 80-100;100-135) for each frame. We dispose over a total of 5271 images for test and SVM gives the following results: a correction rate of 97.31%, a sensitivity of 92.6% and a specificity of 97.67%. Table 1 shows the confusion matrix.

Most of false alarms are isolated, so a tracking over frames could improve detection results. The false negatives are produced exactly by images where rolling pads are merged with the center rail (center of turnout).

VII. Conclusion and Future works

In this paper, a new generic and robust system for railway track and turnout detection is described. In our approach, an edge detection is used with simple geometric constraints provided by the rail characteristics. This leads to a reduction of the computational load and error rate in railway track detection.

While the train is moving, the area between both railway tracks is analyzed to detect turnouts at a distance of $10m$ using the bird-eye view. Results show a very good performance

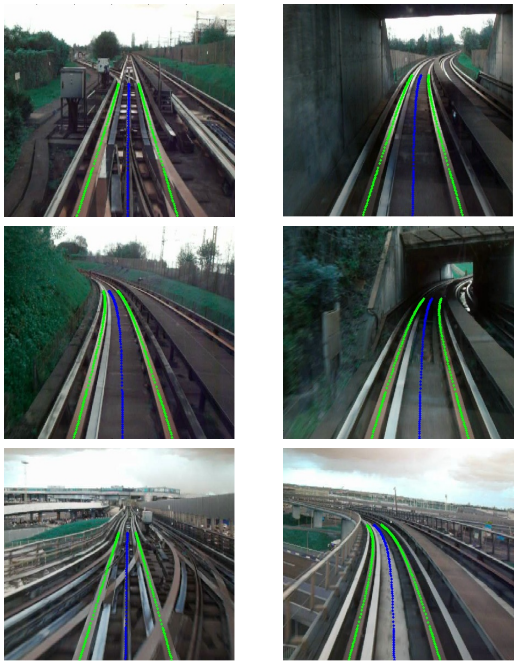


Fig. 10. Results for rails extraction in different situations.

in railway turnout detection by using template matching and HOG.

Future works will include: (i) the search of secondary rails when a turnout is found by changing the approach in RANSAC and (ii) the use of detected railway turnouts for topological trains mapping and localization.

Acknowledgements

Jorge Corsino Espino gratefully acknowledges the support of SIEMENS S.A.S. Infrastructure & Cities, Division Mobility and Logistics and the Mines ParisTech Robotic Center.

References

[1] J. McCall and M. Trivedi, "Video-based lane estimation and tracking for driver assistance: survey, system, and evaluation," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 7, no. 1, pp. 20 –37, march 2006.

[2] V. Kastrinaki, M. Zervakis, and K. Kalaitzakis, "A survey of video processing techniques for traffic applications," *Image and Vision Computing*, vol. 21, pp. 359–381, 2003.

[3] B. Nassu and M. Ukai, "Rail extraction for driver support in railways," in *Intelligent Vehicles Symposium (IV), 2011 IEEE*, june 2011, pp. 83 –88.

[4] F. Maire, "Vision based anti-collision system for rail track maintenance vehicles," in *Advanced Video and Signal Based Surveillance, 2007. AVSS 2007. IEEE Conference on*, sept. 2007, pp. 170 –175.

[5] F. Kaleli and Y. Akgul, "Vision-based railroad track extraction using dynamic programming," in *Intelligent Transportation Systems, 2009. ITSC '09. 12th International IEEE Conference on*, oct. 2009, pp. 1 –6.

[6] F. Maire and A. Bigdeli, "Obstacle-free range determination for rail track maintenance vehicles," in *Control Automation Robotics Vision (ICARCV), 2010 11th International Conference on*, dec. 2010, pp. 2172 –2178.

[7] J. Wohlfeil, "Vision based rail track and switch recognition for self-localization of trains in a rail network," in *Intelligent Vehicles Symposium (IV), 2011 IEEE*, june 2011, pp. 1025 –1030.

[8] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, Jun. 1981. [Online]. Available: http://doi.acm.org/10.1145/358669.358692

[9] R. Brunelli, *Template Matching Techniques in Computer Vision: Theory and Practice*. Wiley Publishing, 2009.

[10] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *In CVPR*, 2005, pp. 886–893.

A.3 Article ITSC 2013

Turnout detection and classification using a modified HOG and template matching

Jorge Corsino Espino[1] and Bogdan Stanciulescu[2]

Abstract— **This paper presents a railway track and turnout detection and turnout classification algorithm. The railway track extraction is based on an edge detection using the width of the rolling pads. This edge detection scheme is then used as an input to the RANSAC algorithm to determine the model of the rails knowing their gauge. The turnout detection scheme is based on the Histogram of Oriented Gradient (HOG) and Template Matching (TM). The turnout classification is based on HOG. The detection results show (i) reliable performance for our railway track extraction scheme; (ii) a correction rate of 97.31 percent for the turnout detection scheme using a Support Vector Machine (SVM) classifier. The turnout classification has correction rate of 98.72 percent using SVM.**

I. Introduction

Trains are considered one of the safest means of transportation. Unfortunately, accidents still often occur due to driver's tiredness, bad weather conditions or other factors. Safety can be improved using intelligent systems that provide additional information about the exact location of the train, its speed and upcoming obstacles. This can be accomplished by installing sensors along the tracks but it can be expensive at a large scale. A more cost efficient approach is to install on-board cameras at the front of the train to support the driver.

Rail recognition and extraction is one of the fundamental tasks in an on-board computer vision system for trains. It can be useful for trains positioning and speed estimation as well as for obstacle detection, determining the region of interest to search obstacles. However, it faces many challenges such as bad illumination and poor visibility in tunnels or under bad weather conditions.

Furthermore, approaching to a turnout, the system has to analyze both directions to determine both tracks as free obstacle path. Railway turnouts are characterized by complex rails extraction scenarios. For this reason, this paper proposes an approach for rails and turnout detection and classification using only image processing to determine a free path in front of the train.

The paper is organized as follows: in Section 2 we describe the state of the art; in Section 3, we give an overview of the proposed system; in Section 4, we detail the proposed system for rails extraction; in Section 5, we present the approach for turnout detection; in Section 6, we present the approach for turnout classification; in Section 7 we present some experimental results. Finally, in Section 8, we summarize the conclusions and give an outlook on future works.

*This work was supported by Mines-ParisTech and SIEMENS S.A.S.

[1]Jorge Corsino Espino is a Phd student in Mathematics and Computer Vision Science, SIEMENS S.A.S. Infrastructure & Cities, Division Mobility and Logistics, 92320 Chatillon, France jorge.corsino-espino@siemens.com

[2]Bogdan Stanciulescu is a researcher and professor with the Robotics center, Mines-ParisTech, 75272 Paris, France bogdan.stanciulescu@mines-paristech.fr

II. Previous work

Some studies [1], [2] show that the lane marking detection techniques, used in intelligent vehicles, can also be applied to the rail detection in certain applications. Both the lane-markings and the rail tracks are represented as a group of lines. Furthermore, the curvatures in both applications tend to have a large radius, barely changing over small distances. However, the textured structure of the track-bed, due to ballast, grass and sleepers, as opposed to the uniform asphalt background of the lane-markings, makes the rails more difficult to be detected. Furthermore, the reflective nature of the steel causes larger variations due to shadows and changes in illumination.

Previous works have exploited the aforementioned characteristics of the traditional railway tracks to detect them as detailed in [3].

In [4], the switches at the railway crossings are detected. The eigenvalues of the gradients computed over image patches are used to identify homogeneous and textured regions. The former is discarded, while the latter is filtered according to the gradient angles. Only a restricted area of the image is examined to reduce the computational overhead. The calculation of the covariance and the eigenvalues can be expensive at a large scale.

In this paper, we propose a rail detection method for rubber-tyred trains, using the RANdom SAmple Consensus (RANSAC) [5] algorithm iteratively, selecting the best model from frame to frame. We test our method on challenging sequences with several parallel rails, richly textured background as well as illumination variations due to shadows and tunnels. At the same time, the system detects turnout sections over the rails to change the approach of RANSAC. In order to detect turnouts the system analyzes the area between rails, searching for the presence of a guidance rail of turnouts.

III. Overview of the method

A. The environment

Rubber-tyred metro is a rapid transportation system that uses a mix of road and rail technology. The vehicle has wheels which run on rolling pads inside guide bars for traction, as well as the traditional railway steel wheels with deep flanges on steel tracks for guidance.

The environment is particularly challenging in train systems because of rail's equipment and tunnels. Besides rubber-tyred metro has a richly textured track-bed as shown in Fig. 1. The complexity of this track-bed makes rail detection difficult due to the large quantity of straight segments and the shadows of the guidance bars produced on the pads. As seen in Fig. 1, the rolling pads and other structures are close to each other. This task is more complex at night or on cloudy days because of rails' color variation.

B. Rails Geometry

Rails geometry describes tracks' layouts with specific alignments, curvature warp, rail gauge, etc. In our case the principal characteristics are:

- Minimum horizontal curve radius: $20m$
- Minimum vertical curve radius: $200m$
- Maximum slope: 12%
- Width of rolling pads $w = 250mm$
- Standard rails gauge $g = 1650mm$

Movement tolerance of the rail vehicle

- Banking: from 0% to 13% $(0,55\%/m)$
- Roll: $< 2\,^{\circ}/s$
- Pitch: $< 2\,^{\circ}/s$
- Yaw:$+/-3\,^{\circ}/s$

Turnouts are used to divide or merge tracks into two or more tracks. The purpose of crossings is to allow tracks to intersect at the same level. In our case, turnouts are equipped with a central switch to guide the train over the turnout. This switch is like a central rail of guidance. An example of this kind of turnout is shown in Fig. 1.

Knowing these characteristics we can assume that the track-bed is planar in the camera field of view. Computing an inverse rectifying homography that maps the center camera image to a "bird's-eye" point of view. Formally, we chose a train-centric coordinate system as follows: $+Z$ is forward, $+X$ is right and $+Y$ is up. The curve $f(z)$ representing the center of the rails can be modelled as a quadratic function in the ground plane such as: $f(z) = az^2 + bz + c$. Given rails' width (w), the standard gauge (g) and the inverse rectifying homography (H_r) [6] for the trackbed where $Y = 0$, we can immediately derive the locations X of left and right rails for any Z.

C. Camera

A forward-facing camera was installed behind the wind shield at a height h of $2.78m$ above the trackbed surface and with a tilt angle of $\theta = 10\,^{\circ}$. Our sequences consist of videos taken at $30fps$ recorded by a camera with $6.2mm$ focal length and $1/3inch$ sensor size. The image resolution is 480 by 640 pixels.

D. Rails and turnout detection and turnout classification approach

Although the goal of this work is turnout detection and classification, it is crucial to detect rails at first. It starts by detecting the center of the rails at the bottom of the image. Then the image's gradients are analyzed to find the rails candidates. These points $p(u, v)$ are then converted with the Homography (H_r) to 3D points $P(X, Y, Z)_{Z=0}$. It is in this space where we fit the best model of the rails. For this purpose we propose to use RANSAC. In order to reduce the number of edges and keeping only the rails, other operations are exploited: masking, binarization, gradient angle analysis and filtering.

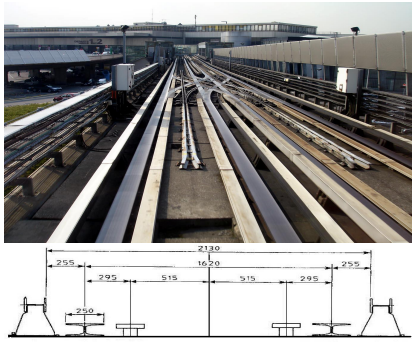


Fig. 1. Driver's view with a fixed camera placed behind the wind-shield and vertical section of track bed and its distances.

Once rails have been detected, turnout detection is launch, searching for central rail of guidance. Then we classify the turnout between 5 classes. Our algorithm is organized as follows:

1) Center of rails detection
2) Masking
3) Gradient analysis
4) Homography: $p(u, v)$ to $P(X, Y, Z)$
5) RANSAC
   a) Left rail RANSAC
   b) Right rail RANSAC
   c) Central parabola
6) Vanishing point of rails
7) Turnout detection (SVM)
   a) Template matching
   b) HOG
8) Turnout Classification (SVM)
   a) HOG of bird's-eye view sub-image

IV. Proposed approach for Rails Detection

A. Edged-based rails detection

The most common techniques used for road lane and rail detection are edge extraction and lines detection in the images. Many of the existing approaches rely on empirically defined thresholds and fine-tuned parameters. We propose to develop a more generic approach to find rails' edges, searching for local maxima and minima gradient separated by a width of w. Before applying the edge detection, it is important to filter the image to smooth out any noise arising

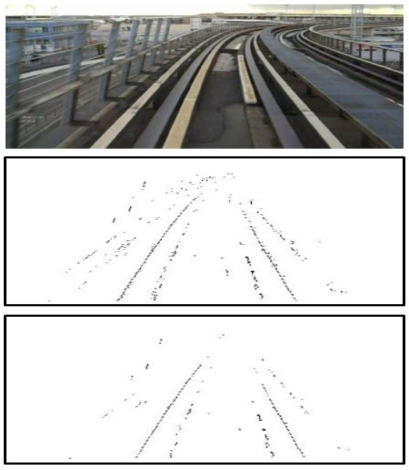


Fig. 2. Process of extracting edges rails candidates.First image is the input image. Second one shows the rails candidates obtained with the gradient analysis. Third image is the filtred image by neighbouring 4x4.

Fig. 3. Example of mask.

from the image capturing. This noise can lead to the detection of false edges.

We can generate a maximum amount of edges as rails candidates, but to reduce the computational overhead, only almost parallel gradients, (having similar edge orientation) are kept. Therefore, horizontal and vertical gradients are analyzed. We generate a binary image with the center position of edges found, considering them as rails candidates. This image is then filtered by a neighbouring mask of size 4x4. As seen in Fig. 2, the number of edges is reduced to only a few, filtering a large amount of erroneous edges and simplifying next steps.

B. Masking

In order to speed up the algorithm and to reduce the number of edges we apply a mask over the image. This mask defines a region of interest with the track's shape but with a bigger width. To place appropriately this mask, it is necessary to know the rails' position at the bottom of the image and at a far distance. For that, the vanishing point (VP) of the rails is calculated and used as the end point of our mask. The VP of the rails is obtained by analyzing the tangent to detected rails at its maximum curve. As remainder, the tangent of a curve at a point a is given by the equation (1)

$$T(x) = f'(a)(x - a) + f(a) \tag{1}$$

An example of the mask can be seen in Fig. 3. For the first frames, the mask is not applied until we have determined a good rails detection. This mask reduces de computational cost and the probability to find erroneous rails.

C. Fitting the rails

From the precedent binary image, we obtain the retro-projection points $P(X, Y, Z)$ for every rails candidate and store them in vector M to extract the rails in the rolling plane. For this purpose we decided to apply an iterative procedure to reduce error detection probability. We firstly use RANSAC to detect left and right rail independently. RANSAC is a robust estimation technique based on the principle of hypotheses, generation and verification, fitting models in the presence of many data outliers. For each search, we impose some conditions to the RANSAC algorithm as stated below:

1) Randomly select three points at different distances Z reducing the probability of a wrong model and the number of iterations.
2) Estimate the model m fitting these three initial inliers.
3) Check model: Verify that the radius of curvature corresponds to the rails' layout.
4) Find the number of M's data items that fit the model with a tolerance t, avoiding to accept other track-pad structure as a rail.
5) Calculate the model's error considering its rail's partner. As the rail's gauge is constant, we can calculate the model's error found at a distance g.
6) Repeat 1-5 L times

To calculate the model's error, we consider the opposite track. Using two Gaussians separated by a width gauge (g), and centered over the rails all along Z distances, we calculate the error of all M's points to the Gaussians. The best fitted model will have the minimum error.

Considering the presence of vertical curves ($slopes$) and vibrations of the rail's vehicle, the homography (H_r) can produce errors in converting $p(u, v)$ to $P(X, Y, Z)$. This is why we decided to analyze each rail separately to obtain the rails' central function curve. From each RANSAC, we keep the inliers and then we translate them to the rails' center. Once the translation is done, a new model is obtained by weighted least squares (equation 2). Weights depend on distance Z. Using this translation of both inliers to the rails' center, the algorithm can lead to small changes in the flat world assumption. The process of fitting the model is shown in Fig. 4.

For the next frame, as the position of rails is known, we can pre filtrate the new M vector to reduce the RANSAC process. This rail extraction method is efficient and robust. An example of edge rails tracker working accurately is shown in Fig. 9.

$$WSS = \sum_{i=1}^{n} W_i r_i^2 \quad , Wi = \frac{1}{D_{z_i}}, \quad r_i^2 = y_i - f(x_i, \beta) \tag{2}$$

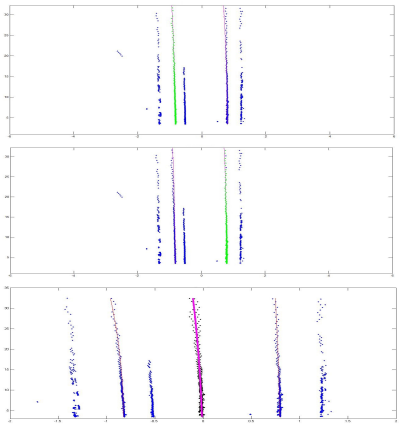


Fig. 4. Retro-projection of edges to fit models with RANSAC. First image shows the result of RANSAC for the left rail. Second is the result for right rail. Third image shows the translation of both inliers

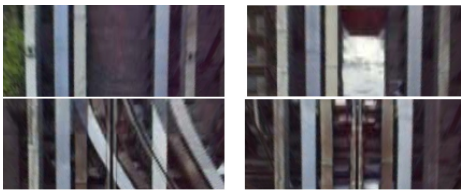


Fig. 5. Bird-eye view images of rails at $10m$. On the top, bird's-eye image of simple rails. At the bottom, with turnouts.

V. Proposed approach for Turnout Detection

The use of bird-eye view makes the detection of turnouts easier because tracks appear parallel, have constant width and gauge, and are mainly vertical. Besides, every pixel appears in world coordinates.

For rubber-tyred metro, in presence of a turnout, there is a central rail to guide the train over it.

The purpose of turnout detection is to determine their position before reaching them. Knowing its emplacement, rails detection system can search for the new pair of rails going to right of left. Once rails are detected we get the bird's-eye view of rails at a distance Z between $10m$ and $12m$ with a lateral offset of $1m$, obtaining an unwrap image of $2x2m$. Some examples of rails bird's-eye view at $10m$ are shown in Fig. 5. We consider to analyze the center of the unwrap image, because in our case all turnouts have a central rail. Hence we propose two techniques to find this rail: Template image matching and Histogram of oriented gradients (HOG).

A. Template matching

Template image matching [7] consist of finding a small part of an image which matches a template image. A basic problem of template matching is to determine the position of

Fig. 6. At left the template image for center rail detection. At right an example of a center rail in bird's-eye view

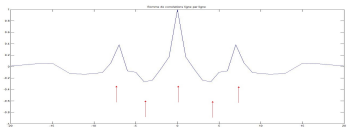


Fig. 7. Auto cross correlation of the template.

a given pattern in an image. In our case, the position of the center rail in the bird's-eye view image is almost constant. Another problem is to determine how much the image and template's brightness can vary due to lighting and exposure conditions. Therefore images have to be normalized first. This can be done at every step by subtracting the mean and dividing by the standard deviation.

Given the center rail's template t as shown in Fig. 6, the idea of the algorithm is to calculate normalized cross-correlations of template and sub-image line per line. Then correlation is calculated as a sum of whole lines correlations. Fig. 7 shows the result of the template's auto-cross correlation. This result is characterized by 3 peaks and 2 valleys.

Taking as correlation feature the value of the center peak (cross-correlation coefficient), we can distinguish between images with and without turnout. But if we take the sum of these 3 peaks and the 2 valleys (equation 3) the discrimination power between two classes is bigger, as shown in figures 8 and 9. Hence, we can use this sum as correlation feature to determine the presence of a central rail (turnout) or not.

$$Corr_feature = \sum_{i=1}^{3} peak + \sum_{i=1}^{2} valley \qquad (3)$$

B. HOG

Histogram of Oriented Gradients [8] is a feature descriptor that gives local shape information often well described by the distribution of intensity gradients or edge directions. The technique counts occurrences of gradient orientation in localized portions of an image. Each pixel votes for an orientation according to the closest bin in the range. Our bins range is defined from 0 do 180 with 5 bins centred in $[7.5; 50; 90; 115; 157.5]$. We use just one cell for HOG algorithm because the bird's-eye view has a reduced size and we only analyze a small area enclosed by the rails. The optimum 3x3 mask (equation 4) to obtain gradients is based in [9].

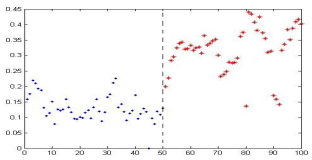


Fig. 8. Maximum value of correlation peak for 50 images without turnout and 50 with turnout.

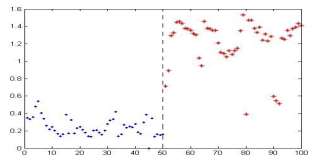


Fig. 9. Correlation feature extracted for 50 images without turnout and 50 with turnout.

$$M = \begin{bmatrix} -0.112737 & 0 & 0.112737 \\ -.274526 & 0 & .274526 \\ -0.112737 & 0 & 0.112737 \end{bmatrix} \quad (4)$$

As expected, in presence of a turnout, HOG reveals vertical gradient of the center rail and the spatial distribution of bins is centered in the 3rd bin. Otherwise all bins have a similar amplitude. In Fig. 10 we can see the distribution of bin's amplitudes for images with and without turnout.

VI. Proposed approach for Turnout Classification

Facing and trailing are terms used to describe railway turnouts in respect to whether they are divergent or convergent respectively. When a train traverses a turnout in a facing direction, it may diverge onto either of the two routes. When travelled in a trailing direction, the two routes converge onto each other. With the purpose of classify turnouts we decided to analyse the whole bird's eye view image using HOG. Knowing the distribution of gradients' orientations in this image we can classify the turnout in 5 classes.

1) Arriving/Leaving a turnout (just central rail is presented)
2) Right-hand facing turnout.
3) Left-hand facing turnout.
4) Left-hand trailing turnout.
5) Right-hand trailing turnout.

Our bins' range is defined from 0 do 180 degrees with 5 bins centred in $[25; 67.5; 90; 112.5; 155]$. Because of unwrap characteristics image, the orientation of leaving/arrival rails in turnout have the same angle. In our case, we only kept the 3 central bins for our HOG feature. Analyzing the shape of turnouts in the bird's-eye view we chose 12 non overlapped cells.

As classifier we propose to use SVM multi class, building 5 binary classifiers using the method one-versus-all.

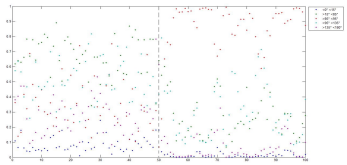


Fig. 10. Distribution of HOG bin's amplitude for 50 images without turnout and 50 with turnout.

Fig. 11. Shape of 5 classes of turnout to classify in the bird's-eye view image.

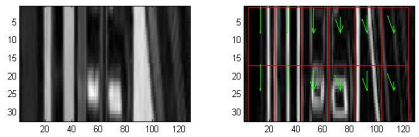


Fig. 12. HOG(12 cells,3 central bins) for right-hand trailing turnout.

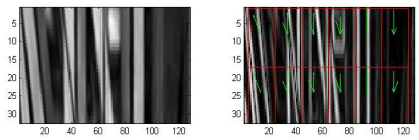


Fig. 13. HOG(12 cells,3 central bins) for left-hand facing turnout.

VII. Results

Our algorithm was tested using different videos of the metro line OrlyVal in a cloudy day. In these videos the train rolls over 6 different turnouts. As explained, it is necessary to extract rails before detecting turnouts. The mean of rails' maximum distance detected is 25 meters. This distance is limited by the low resolution of the camera. Fitting results under different circumstances are shown in Fig. 14. Green lines present the fitted model for each rail and. All along the videos, rails extraction is done accurately.

For turnout detection we use as a classifier a Support Vector Machine (SVM) trained with less than $1/3$ of data base images. As feature to detect the presence of central switch we use a correlation feature and 3 bins of HOG $[15-80; 80-100; 100-135]$ for each frame. This feature is the input to the SVM classifier. We dispose over a total of 5271 images for test and SVM gives the following results: a correction rate of 97.31%, a sensitivity of 92.6% and a specificity of 97.67%. Table I shows the confusion matrix. Most of false alarms are isolated, so a tracking over frames could improve detection results. The false negatives are produced exactly by images where rolling pads are merged with the center rail (center of turnout). Finally, for turnout

TABLE I

CONFUSION MATRIX FOR TURNOUT DETECTION

		Output		
		Positive	Negative	Total
Input	Positive	526	32	568
	Negative	110	4603	4713
	Total	636	4635	5271

TABLE II

CONFUSION MATRIX FOR TURNOUT CLASSIFICATION

		Output					
		c.1	c.2	c.3	c.4	c.5	Total
Input	c.1	3504	0	1	0	1	3506
	c.2	14	41	0	0	1	56
	c.3	1	0	10	0	0	11
	c.4	7	1	0	10	1	19
	c.5	18	2	0	0	63	83
	Total	3544	44	11	10	66	3675

classification we use a SVM multi class classifier, using the structure one versus all[10]. The feature a modified HOG of bird's-eye view sub-image. We use less than $1/3$ of turnout images to train SVM. As our database is reduced, we decided to add to class 1 all the frames without turnout as the global shape of pads is the same (2 vertical pads). In this case, we obtain a total correct rate of 98.72 percent for all classes. Table II shows its confusion matrix. Most of errors between turnout classes are due to double turnouts, where tracks merge and separate in the same turnout. Theses errors are isolated so as before, a tracking over frames can suppress them increasing the accuracy. In Fig. 14 we can see different turnout detection and its classification. Yellow lines represents the area analyzed in bird's-eye view to detect and classify turnouts.

VIII. CONCLUSION AND FUTURE WORKS

In this paper a new generic and robust system for railway track and turnout detection and turnout classification is described. In our approach, an edge detection is used with simple geometric constraints provided by the rail characteristics. This leads to a reduction of the computational load and error rate in railway track detection.

While the train is moving, just a section of the area between both railway tracks is analyzed to detect turnouts using its bird's-eye view. Once a turnout is detected it is classified using HOG as input for a SVM multi-class classifier. Results show a very good performance in railway turnout detection and classification by using template matching and HOG.

Future works will include: (i) the search of secondary rails when a facing turnout is found by changing the approach in RANSAC and (ii) the use of detected railway turnouts for topological trains mapping and localization.

ACKNOWLEDGEMENTS

Jorge Corsino Espino gratefully acknowledges the support of SIEMENS S.A.S. Infrastructure & Cities, Division Mobility and Logistics in Chatillon, specially to Philippe Forin, and the Mines ParisTech Robotic Center.

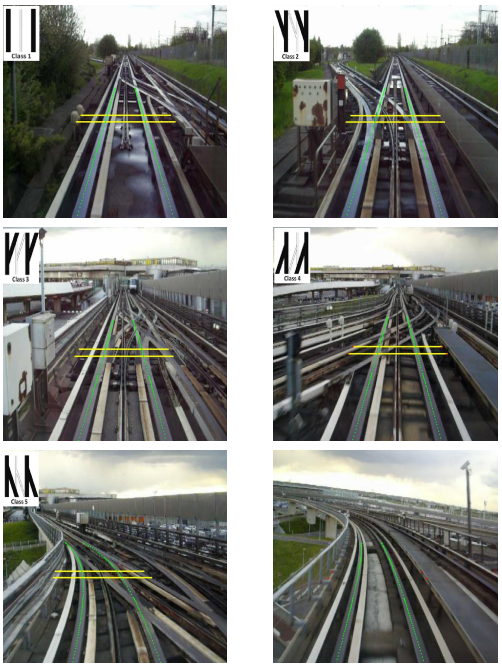


Fig. 14. Results for rails extraction, turnout detection and classification under different situations. $Class$ 1; $Class$ 2; $Class$ 3; $Class$ 4; $Class$ 5; and $None\ turnout$

REFERENCES

[1] J. McCall and M. Trivedi, "Video-based lane estimation and tracking for driver assistance: survey, system, and evaluation," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 7, no. 1, pp. 20 –37, march 2006.

[2] V. Kastrinaki, M. Zervakis, and K. Kalaitzakis, "A survey of video processing techniques for traffic applications," *Image and Vision Computing*, vol. 21, pp. 359–381, 2003.

[3] J. Espino and B. Stanciulescu, "Rail extraction technique using gradient information and a priori shape model," in *Intelligent Transportation Systems (ITSC), 2012 15th International IEEE Conference on*, 2012, pp. 1132–1136.

[4] J. Wohlfeil, "Vision based rail track and switch recognition for self-localization of trains in a rail network," in *Intelligent Vehicles Symposium (IV), 2011 IEEE*, june 2011, pp. 1025 –1030.

[5] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, Jun. 1981.

[6] A. Agarwal, C. V. Jawahar, and P. J. Narayanan, "A Survey of Planar Homography Estimation Techniques," Centre for Visual Information Technology, International Institute of Information Technology, Hyderabad, India, Tech. Rep., 2005.

[7] R. Brunelli, *Template Matching Techniques in Computer Vision: Theory and Practice*. Wiley Publishing, 2009.

[8] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *In CVPR*, 2005, pp. 886–893.

[9] S. Ando, "Consistent gradient operators," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 22, no. 3, pp. 252–265, 2000.

[10] C.-W. Hsu and C.-J. Lin, "A comparison of methods for multiclass support vector machines," 2002.

Bibliographie

[Abramson *et al.* 2007] Yotam Abramson, Bruno Steux et Hicham Ghorayeb. *Yet Even Faster (YEF) real-time object detection.* International journal of intelligent systems technologies and applications, vol. 2, no. 2, pages 102–112, 2007. (Cité en page 103.)

[Aksoy *et al.* 2004] M.S. Aksoy, O. Torkul et I.H. Cedimoglu. *An industrial visual inspection system that uses inductive learning.* Journal of Intelligent Manufacturing, vol. 15, no. 4, pages 569–574, 2004. (Cité en page 78.)

[Ando 2000] S. Ando. *Consistent gradient operators.* Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 22, no. 3, pages 252–265, 2000. (Cité en page 82.)

[Arrospide *et al.* 2012] Jon Arrospide, Luis Salgado et Javier Marinas. *HOG-like gradient-based descriptor for visual vehicle detection.* In Intelligent Vehicles Symposium (IV), 2012 IEEE, pages 223–228. IEEE, 2012. (Cité en page 106.)

[Ballard 1981] Dana H Ballard. *Generalizing the Hough transform to detect arbitrary shapes.* Pattern recognition, vol. 13, no. 2, pages 111–122, 1981. (Cité en pages 18, 32, 33 et 44.)

[Barczak *et al.* 2008] Andre LC Barczak, Martin J Johnson et Chris H Messom. *Empirical evaluation of a new structure for adaboost.* In Proceedings of the 2008 ACM symposium on Applied computing, pages 1764–1765. ACM, 2008. (Cité en page 103.)

[Bargeton 2009] Alexandre Bargeton. *Fusion multi-sources pour l'interprétation d'un environnement routier.* PhD thesis, École Nationale Supérieure des Mines de Paris, 2009. (Cité en pages 48 et 50.)

[Bay *et al.* 2006] Herbert Bay, Tinne Tuytelaars et Luc Van Gool. *Surf : Speeded up robust features.* In Computer Vision–ECCV 2006, pages 404–417. Springer, 2006. (Cité en page 102.)

[Bdiri *et al.* 2009] Taoufik Bdiri, Fabien Moutarde, Nicolas Bourdis et Bruno Steux. *Adaboost with "Keypoint Presence Features" for Real-Time Vehicle Visual Detection.* CoRR, vol. abs/0910.1273, 2009. (Cité en pages 103 et 104.)

[Bertozzi *et al.* 2000] Massimo Bertozzi, Alberto Broggi et Alessandra Fascioli. *Vision-based intelligent vehicles State of the art and perspectives.* Robotics and Autonomous systems, vol. 32, no. 1, pages 1–16, 2000. (Cité en pages 41 et 98.)

[Beucher 1994] Serge Beucher. *Watershed, hierarchical segmentation and waterfall algorithm.* In Mathematical morphology and its applications to image processing, pages 69–76. Springer, 1994. (Cité en page 45.)

[Beugin 2006] Julie Beugin. *Contribution à l'évaluation de la sécurité des systèmes complexes de transport guidé.* PhD thesis, Université de Valenciennes et du Hainaut-Cambresis, 2006. (Cité en page 43.)

[Bosch *et al.* 2007] Anna Bosch, Andrew Zisserman et Xavier Munoz. *Representing shape with a spatial pyramid kernel.* In Proceedings of the 6th ACM international conference on Image and video retrieval, pages 401–408. ACM, 2007. (Cité en page 91.)

[Bouguet 2006] Jean-Yves Bouguet. *Complete Camera Calibration Toolbox for Matlab*, 2006. (Cité en page 28.)

[Boulaassal 2010] Hakim Boulaassal. *Segmentation et modélisation géométrique de façades de bâtiments à partir de relevés laser terrestres.* PhD thesis, Université de Strasbourg, 2010. (Cité en page 62.)

[Broggi *et al.* 2004] A. Broggi, P. Cerri et P.C. Antonello. *Multi-resolution vehicle detection using artificial vision.* In Intelligent Vehicles Symposium, 2004 IEEE, pages 310–314, June 2004. (Cité en page 106.)

[Broggi 1995] Alberto Broggi. *Robust real-time lane and road detection in critical shadow conditions.* In Computer Vision, 1995. Proceedings., International Symposium on, pages 353–358. IEEE, 1995. (Cité en page 41.)

[Brunelli 2009] Roberto Brunelli. Template matching techniques in computer vision : Theory and practice. Wiley Publishing, 2009. (Cité en page 78.)

[Canny 1986] John Canny. *A computational approach to edge detection.* Pattern Analysis and Machine Intelligence, IEEE Transactions on, no. 6, pages 679–698, 1986. (Cité en pages 32 et 43.)

[Carbonetto *et al.* 2008] Peter Carbonetto, Gyuri Dorkó, Cordelia Schmid, Hendrik Kück et Nando De Freitas. *Learning to recognize objects with little supervision.* International Journal of Computer Vision, vol. 77, no. 1-3, pages 219–237, 2008. (Cité en page 100.)

[Cerri & Grisleri 2005] P. Cerri et P. Grisleri. *Free Space Detection on Highways using Time Correlation between Stabilized Sub-pixel precision IPM Images.* In Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on, pages 2223–2228, April 2005. (Cité en page 46.)

[Chang & Lin 2011] Chih-Chung Chang et Chih-Jen Lin. *LIBSVM : a library for support vector machines.* ACM Transactions on Intelligent Systems and Technology (TIST), vol. 2, no. 3, page 27, 2011. (Cité en page 84.)

[Chanton 2008] Richard Chanton. *Train video positioning.* Institution of Railway Signal Engineers ASPECT, 2008. (Cité en page 68.)

[Chen & Wang 2006] Qiang Chen et Hong Wang. *A Real-time Lane Detection Algorithm Based on a Hyperbola-Pair Model.* In Intelligent Vehicles Symposium, 2006 IEEE, pages 510–515, 2006. (Cité en page 53.)

[Choi *et al.* 2012] H-C Choi, J-M Park, W-S Choi et S-Y Oh. *Vision-based fusion of robust lane tracking and forward vehicle detection in a real driving environment.* International Journal of Automotive Technology, vol. 13, no. 4, pages 653–669, 2012. (Cité en pages 53 et 57.)

[Dalal & Triggs 2005] Navneet Dalal et Bill Triggs. *Histograms of Oriented Gradients for Human Detection.* In In CVPR, pages 886–893, 2005. (Cité en pages 81, 82, 90 et 91.)

[Delacour *et al.* 2005] H Delacour, A Servonnet, A Perrot, JF Vigezzi et JM Ramirez. *La courbe ROC (receiver operating characteristic) : principes et principales applications en biologie clinique.* In Annales de biologie clinique, volume 63, pages 145–154, 2005. (Cité en page 85.)

[Dharpure *et al.* 2013] Jaydeo K. Dharpure, M. B. Potdar et Manoj Pandya. *Article : Counting Objects using Convolution based Pattern Matching Technique.* International Journal of Applied Information Systems, vol. 5, no. 8, pages 14–19, June 2013. Published by Foundation of Computer Science, New York, USA. (Cité en page 78.)

[Dominique & Eric 1998] Meize Dominique et Brassart Eric. *Calibration du capteur de vision omnidirectionnelle : SYCLOP.* In Diplome d'Etudes Approfondies, Controle des Systèmes. Université de Compiègne, 1998. (Cité en page 24.)

[Dooze *et al.* 1998] D. Dooze, L. Khoudour et C. Vieren. *Obstacle detection in front of automatic trains by linear stereo vision.* In Computers in Railways VI, volume The Built Environment volume 37. C. A. BREBBIA, Wessex Institute of Technology, United Kingdom,, 1998. (Cité en pages 17 et 97.)

[Espino *et al.* 2011] Jorge Corsino Espino, Bruno Steux et OE Hamzaoui. *Safe navigating system for indoor environments.* In Automation, Robotics and Applications (ICARA), 2011 5th International Conference on, pages 419–423. IEEE, 2011. (Cité en page 20.)

[Européene 2003a] Commission Européene. *Norme Européene NF EN 50126 Applications ferroviaires - Spécification et démonstration de la fiabilité, de la disponibilité, de la maintenabilité et de la sécurité*, Mai 2003. (Cité en page 4.)

[Européene 2003b] Commission Européene. *Norme Européene NF NE 50129 Application Ferroviaires Systémes de signalisation, de télécommunication et de Traitement Systèmes électroniques de sécurité pour la signalisation*, Mai 2003. (Cité en page 4.)

[Everingham *et al.*] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn et A. Zisserman. *The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results.* http ://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html. (Cité en pages 100 et 101.)

[Fischler & Bolles 1981] Martin A. Fischler et Robert C. Bolles. *Random sample consensus : a paradigm for model fitting with applications to image analysis and automated cartography.* Commun. ACM, vol. 24, no. 6, pages 381–395, jun 1981. (Cité en page 57.)

[Fritsch *et al.* 2013] Jannik Fritsch, Tobias Kuehnl et Andreas Geiger. *A New Performance Measure and Evaluation Benchmark for Road Detection Algorithms.* In International Conference on Intelligent Transportation Systems (ITSC), 2013. (Cité en page 39.)

[Gangnet *et al.* 1984] Michel Gangnet, Didier Perny et Philippe Coueignoux. *Perspective mapping of planar textures.* Computers & Graphics, vol. 8, no. 2, pages 115–123, 1984. (Cité en pages 16 et 43.)

[Garcia *et al.* 2005] J.J. Garcia, C. Losada, F. Espinosa, J. Urena, A. Hernandez, M. Mazo, C. De Marziani, A. Jimenez, E. Bueno et F. Alvarez. *Dedicated smart IR barrier for obstacle detection in railways.* In Industrial Electronics Society, 2005. IECON 2005. 31st Annual Conference of IEEE, page 6, 2005. (Cité en page 3.)

[Geistler & Bohringer 2004] A Geistler et F Bohringer. *Detection and classification of turnouts using eddy current sensors.* Publication of : WIT Press, 2004. (Cité en page 73.)

[Gschwandtner *et al.* 2010] Michael Gschwandtner, Wolfgang Pree et Andreas Uhl. *Track detection for autonomous trains.* In Advances in Visual Computing, pages 19–28. Springer, 2010. (Cité en pages 42, 43, 48 et 51.)

[Hamdoun 2010] Omar Hamdoun. *Détection et ré-identification de piétons par points d'intérêt entre caméras disjointes.* PhD thesis, École nationale supérieure des mines de Paris, 2010. (Cité en page 102.)

[Hancock *et al.* 1998] John Hancock, Martial Hebert et Chuck Thorpe. *Laser intensity-based obstacle detection.* In Intelligent Robots and Systems, 1998. Proceedings., 1998 IEEE/RSJ International Conference on, volume 3, pages 1541–1546. IEEE, 1998. (Cité en page 19.)

[Harris & Stephens 1988] Chris Harris et Mike Stephens. *A combined corner and edge detector.* In Alvey vision conference, volume 15, page 50. Manchester, UK, 1988. (Cité en pages 67 et 102.)

[Harzallah 2011] Hedi Harzallah. *Contribution à la localisation et à la reconnaissance d ?objets dans les images.* PhD thesis, Université de Caen, 2011. (Cité en pages 103 et 106.)

[Haykin 1994] Simon Haykin. Neural networks : a comprehensive foundation. Prentice Hall PTR, 1994. (Cité en page 17.)

[Hsu & Lin 2002] Chih-Wei Hsu et Chih-Jen Lin. *A Comparison of Methods for Multiclass Support Vector Machines*, 2002. (Cité en page 90.)

[Jazayeri *et al.* 2011] A. Jazayeri, Hongyuan Cai, Jiang Yu Zheng et M. Tuceryan. *Vehicle Detection and Tracking in Car Video Based on Motion Model.* Intelligent Transportation Systems, IEEE Transactions on, vol. 12, no. 2, pages 583–595, June 2011. (Cité en page 98.)

[Jung & Kelber 2004] Claudio Rosito Jung et Christian Roberto Kelber. *A robust linear-parabolic model for lane following.* In Computer Graphics and Image Processing, 2004. Proceedings. 17th Brazilian Symposium on, pages 72–79. IEEE, 2004. (Cité en page 57.)

[Kaske *et al.* 1997] Axel Kaske, Didier Wolf et Rene Husson. *Lane boundary detection using statistical criteria.* In International Conference on Quality by Artificial Vision, QCAV9, pages 28–30. Citeseer, 1997. (Cité en page 57.)

[Kass *et al.* 1988] Michael Kass, Andrew Witkin et Demetri Terzopoulos. *Snakes : Active contour models.* International journal of computer vision, vol. 1, no. 4, pages 321–331, 1988. (Cité en page 17.)

[Kawamura *et al.* 2011] A. Kawamura, Y. Yoshimitsu, K. Kajitani, T. Naito, K. Fujimura et S. Kamijo. *Smart camera network system for use in railway stations.* pages 85–90, 2011. (Cité en page 3.)

[Kolmogorov & Zabih 2002] Vladimir Kolmogorov et Ramin Zabih. *Multi camera scene reconstruction via graph cuts.* In Computer Vision ECCV 2002, pages 82–96. Springer, 2002. (Cité en page 24.)

[Labayrade *et al.* 2002] Raphael Labayrade, Didier Aubert et J-P Tarel. *Real time obstacle detection in stereovision on non flat road geometry through" v-disparity" representation.* In Intelligent Vehicle Symposium, 2002. IEEE, volume 2, pages 646–651. IEEE, 2002. (Cité en page 97.)

[Lailler 1993] Casimir Lailler. *Guidage par vision artificielle d'un véhicule autonome sur route matérialisée par la signalisation horizontale. Application à la protection des chantiers mobiles d'entretien du réseau routier.* PhD thesis, 1993. (Cité en page 41.)

[Lee 2007] David C Lee. *Boosted Classifier for Car Detection*, 2007. (Cité en pages 103 et 104.)

[Lefaudeux & Nashashibi 2012] Benjamin Lefaudeux et Fawzi Nashashibi. *Real-time visual perception : detection and localisation of static and moving objects from a moving stereo rig.* In ITSC 2012 - 15th International IEEE Conference on Intelligent Transportation Systems, pages 522 – 527, Anchorage, Etats-Unis, Sep 2012. IEEE. (Cité en page 24.)

[Lefaudeux 2013] Benjamin Lefaudeux. *Doctorat ParisTech.* PhD thesis, École nationale supérieure des mines de Paris, 2013. (Cité en page 102.)

[Liu *et al.* 1990] Y. Liu, T.S. Huang et O.D. Faugeras. *Determination of camera location from 2-D to 3-D line and point correspondences.* Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 12, no. 1, pages 28–37, 1990. (Cité en page 24.)

[Louahdi *et al.* 2002] Khoudour Louahdi, Ambellouis Sebastien, Vieren Christophe et Pascal Deloof. *Détection d'obstacles devant les véhicules de transports guidés. Utilisation d'un radar coopératif et d'un stéréoscope optique.* In Instrumentation Mesure Metrologie Capteurs et Traitement du signal pour les transports guidés, volume 2, pages 81–116. Lavosier, 2002. (Cité en pages 17 et 97.)

[Lowe 1999] David G Lowe. *Object recognition from local scale-invariant features.* In Computer vision, 1999. The proceedings of the seventh IEEE international conference on, volume 2, pages 1150–1157. Ieee, 1999. (Cité en page 102.)

[Lowe 2004] David G Lowe. *Distinctive image features from scale-invariant keypoints.* International journal of computer vision, vol. 60, no. 2, pages 91–110, 2004. (Cité en page 67.)

[Mabrouk 2010] Habib Hadj Mabrouk. *Introduction à la sécurité et à l'analyse des risques technologiques et humains.* In Actes du Symposium International sur la Maintenance et la Maîtrise des Risques, 2010. (Cité en pages 4 et 5.)

[Maire & Bigdeli 2010] Frédéric Maire et Abbas Bigdeli. *Obstacle-free range determination for rail track maintenance vehicles.* In Control Automation Robotics & Vision (ICARCV), 2010 11th International Conference on, pages 2172–2178. IEEE, 2010. (Cité en pages 18 et 78.)

[Mar *et al.* 2011] N.S.S. Mar, P.K.D.V. Yarlagadda et C. Fookes. *Design and development of automatic visual inspection system for PCB manufacturing.* Robotics and Computer-Integrated Manufacturing, vol. 27, no. 5, pages 949 – 962, 2011. (Cité en page 24.)

[Massimo *et al.* 2000] Bertozzi Massimo, Broggi Alberto et Fascioli Alessandra. *Vision-based intelligent vehicles : State of the art and perspectives.* Robotics and Autonomous Systems, vol. 32, no. 1, pages 1 – 16, 2000. (Cité en page 1.)

[Meilland 2012] Maxime Meilland. *Cartographie RGB-D dense pour la localisation visuelle temps-réel et la navigation autonome.* PhD thesis, Ecole Nationale Supérieure des Mines de Paris, 2012. (Cité en page 67.)

[Mockel *et al.* 2003] S Mockel, Frank Scherer et Peter F Schuster. *Multi-sensor obstacle detection on railway tracks.* In Intelligent Vehicles Symposium, 2003. Proceedings. IEEE, pages 42–46. IEEE, 2003. (Cité en pages 15 et 16.)

[Muad *et al.* 2004] A.M. Muad, A. Hussain, S.A. Samad, M.M. Mustaffa et B.Y. Majlis. *Implementation of inverse perspective mapping algorithm for the development of an automatic lane tracking system.* In TENCON 2004. 2004 IEEE Region 10 Conference, volume A, pages 207–210 Vol. 1, Nov 2004. (Cité en page 48.)

[Nangtin *et al.* 2005] P. Nangtin, P. Kumhom et K. Chamnongthai. *Video-based obstacle tracking for automatic train navigation.* In Intelligent Signal Processing and Communication Systems, 2005. ISPACS 2005. Proceedings of 2005 International Symposium on, pages 21–24, 2005. (Cité en page 3.)

[NCAP 2013] EURO NCAP. *EURO NCAP advanced : Autonomous Emergency Braking*, Sept 2013. (Cité en pages 2 et 99.)

[Nieto *et al.* 2007a] M. Nieto, L. Salgado, F. Jaureguizar et J. Cabrera. *Stabilization of Inverse Perspective Mapping Images based on Robust Vanishing Point Estimation.* In Intelligent Vehicles Symposium, 2007 IEEE, pages 315–320, June 2007. (Cité en pages 46 et 47.)

[Nieto *et al.* 2007b] Marcos Nieto, Luis Salgado, Fernando Jaureguizar et Julian Cabrera. *Stabilization of inverse perspective mapping images based on robust vanishing point estimation.* In Intelligent Vehicles Symposium, 2007 IEEE, pages 315–320. IEEE, 2007. (Cité en page 29.)

[Parker 2010] Jim R Parker. Algorithms for image processing and computer vision. Wiley. com, 2010. (Cité en page 16.)

[Perrollaz 2008] Mathias Perrollaz. *Détection d'obstacles multi-capteurs supervisée par stéréovision.* PhD thesis, Université Pierre et Marie Curie-Paris VI, 2008. (Cité en page 20.)

[Pomerleau 1995] Dean Pomerleau. *RALPH : Rapidly adapting lateral position handler.* In Intelligent Vehicles' 95 Symposium., Proceedings of the, pages 506–511. IEEE, 1995. (Cité en page 48.)

[Provost 2005] Jean Provost. Transports intelligents, automatisation des lignes de métro. Notes Module d'ETIC, Telecom ParisTech, 2005. (Cité en page 2.)

[Puhlmann *et al.* 2004] I. Puhlmann, S. Schussler et B. Hulin. *Improvements on obstacle detection in the pantograph gauge due to the recognition of steady arms.* In Intelligent Vehicles Symposium, 2004 IEEE, pages 518–521, 2004. (Cité en page 3.)

[Puthon 2013] Anne-sophie Puthon. *Détermination de la vitesse limite par fusion de données cartographiques et vision temps-réel embarquées.* PhD thesis, École Nationale Supérieure des Mines de Paris, 2013. (Cité en page 91.)

[Rodriguez *et al.* 2012] L.A.F. Rodriguez, J.A. Uribe et J.F.V. Bonilla. *Obstacle detection over rails using hough transform.* In Image, Signal Processing, and Artificial Vision (STSIVA), 2012 XVII Symposium of, pages 317–322, Sept 2012. (Cité en pages 44 et 45.)

[Ross 2012] R. Ross. *Track and turnout detection in video-signals using probabilistic spline curves.* In Intelligent Transportation Systems (ITSC), 2012 15th International IEEE Conference on, pages 294–299, 2012. (Cité en page 23.)

[Ruder *et al.* 2003] M Ruder, N Mohler et Faruque Ahmed. *An obstacle detection system for automated trains.* In Intelligent Vehicles Symposium, 2003. Proceedings. IEEE, pages 180–185. IEEE, 2003. (Cité en pages 16 et 64.)

[Salvi *et al.* 2002] Joaquim Salvi, Xavier Armangué et Joan Batlle. *A comparative review of camera calibrating methods with accuracy evaluation.* Pattern recognition, vol. 35, no. 7, pages 1617–1635, 2002. (Cité en page 27.)

[Sasaki *et al.* 1999] A. Sasaki, Y. Tanaka et T. Ichikura. *On-board self-regulating automated vehicle operation system.* In Vehicular Technology Conference, 1999 IEEE 49th, volume 3, pages 2298–2302 vol.3, 1999. (Cité en page 18.)

[Song *et al.* 2008a] Gwang Yul Song, Ki Yong Lee et Joon Woong Lee. *Vehicle detection by edge-based candidate generation and appearance-based classification.* In Intelligent Vehicles Symposium, 2008 IEEE, pages 428–433. IEEE, 2008. (Cité en page 98.)

[Song *et al.* 2008b] Hyun Oh Song, Sara Bolouki et Andrew Yen. *Detection of Stop Signs.* CS 223B Project Report, 2008. (Cité en page 88.)

[Stallkamp *et al.* 2011] Johannes Stallkamp, Marc Schlipsing, Jan Salmen et Christian Igel. *The German traffic sign recognition benchmark : a multi-class classification competition.* In Neural Networks (IJCNN), The 2011 International Joint Conference on, pages 1453–1460. IEEE, 2011. (Cité en page 40.)

[Stanciulescu *et al.* 2009] Bogdan Stanciulescu, Amaury Breheret et Fabien Moutarde. *Introducing new adaboost features for real-time vehicle detection.* arXiv preprint arXiv :0910.1293, 2009. (Cité en page 104.)

[Steux & El Hamzaoui 2010] Bruno Steux et Oussama El Hamzaoui. *tinySLAM : A SLAM algorithm in less than 200 lines C-language program.* In Control Automation Robotics & Vision (ICARCV), 2010 11th International Conference on, pages 1975–1979. IEEE, 2010. (Cité en page 20.)

[Steux 2001] Bruno Steux. *Camellia Library, Image Processing & Computer Vision library.* http ://camellia.sourceforge.net/, 2001. (Cité en page 103.)

[Sturm 2008] Peter Sturm. *Quelques notes pour les cours de Vision par Ordinateur*, 2008. (Cité en page 28.)

[Suttorp & Bucher 2006] Thorsten Suttorp et Thomas Bucher. *Robust vanishing point estimation for driver assistance.* In Intelligent Transportation Systems Conference, 2006. ITSC'06. IEEE, pages 1550–1555. IEEE, 2006. (Cité en page 35.)

[Tian *et al.* 2006] Min Tian, Fuqiang Liu, Wenhong Zhu et Chao Xu. *Vision based lane detection for active security in intelligent vehicle.* In Vehicular Electronics and Safety, 2006. ICVES 2006. IEEE International Conference on, pages 507–511. IEEE, 2006. (Cité en page 44.)

[Tsai *et al.* 2007] Luo-Wei Tsai, Jun-Wei Hsieh et Kuo-Chin Fan. *Vehicle detection using normalized color and edge map.* Image Processing, IEEE Transactions on, vol. 16, no. 3, pages 850–864, 2007. (Cité en page 98.)

[Tzomakas & von Seelen 1998] Christos Tzomakas et Werner von Seelen. *Vehicle detection in traffic scenes using shadows.* In IR-INI, INSTITUT FUR NUEROINFORMATIK, RUHR-UNIVERSITAT. Citeseer, 1998. (Cité en page 98.)

[Vazquez *et al.* 2004] J. Vazquez, M. Mazo, J.L. Lazaro, C.A. Luna, J. Urena, J.J. Garcia, J. Cabello et L. Hierrezuelo. *Detection of moving objects in railway using vision.* In Intelligent Vehicles Symposium, 2004 IEEE, pages 872–875, 2004. (Cité en page 3.)

[Vedaldi & Fulkerson 2008] A. Vedaldi et B. Fulkerson. *VLFeat : An Open and Portable Library of Computer Vision Algorithms.* `http://www.vlfeat.org/`, 2008. (Cité en page 106.)

[Verhille 2007] Jean-Noël Verhille. *Représentation énergétique macroscopique du métro VAL 206 et structures de commande déduites par inversion.* PhD thesis, Arts et Métiers ParisTech, 2007. (Cité en page 21.)

[Viola & Jones 2001] Paul Viola et Michael Jones. *Rapid object detection using a boosted cascade of simple features.* In Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on, volume 1, pages I–511. IEEE, 2001. (Cité en pages 102 et 103.)

[Voisin *et al.* 2004] Vincent Voisin, Bruno Emile et Manuel Avila. *Détection et suivi de marquages de bords de voie par vision artificielle.* Journées Jeunes Chercheurs du Laboratoire Vision et Robotique, 2004. (Cité en page 41.)

[Wang *et al.* 2007] Kunfeng Wang, Hua Huang, Yuantao Li et Fei-Yue Wang. *Research on Lane-Marking line based camera calibration.* In Vehicular Electronics and Safety, 2007. ICVES. IEEE International Conference on, pages 1–6. IEEE, 2007. (Cité en page 30.)

[Weiner 1973] Peter Weiner. *Linear pattern matching algorithms.* In Switching and Automata Theory, 1973. SWAT'08. IEEE Conference Record of 14th Annual Symposium on, pages 1–11. IEEE, 1973. (Cité en page 18.)

[Wohlfeil 2011] J. Wohlfeil. *Vision based rail track and switch recognition for self-localization of trains in a rail network.* In Intelligent Vehicles Symposium (IV), 2011 IEEE, pages 1025 –1030, june 2011. (Cité en pages 74 et 75.)

[Xavier 2003] Clady Xavier. *Contributions à la navigation autonome d'un véhicule automobile par vision.* PhD thesis, Université Blaise Pascal de Clermont-Ferrand, 2003. (Cité en page 98.)

[Yamashita *et al.* 1996] Hiroshi Yamashita, Jun Nakamoto, Yasushi Koyama et Masanobu Sato. *Development of Railway Obstacle Detection System.* Mitsubishi Heavy Industries, Ltd. Technical Review, vol. 33, no. 1, 1996. (Cité en page 17.)

[Zaklouta & Stanciulescu 2011] Fatin Zaklouta et Bogdan Stanciulescu. *Warning traffic sign recognition using a HOG-based Kd tree.* In Intelligent Vehicles Symposium (IV), 2011 IEEE, pages 1019–1024. IEEE, 2011. (Cité en page 77.)

[Zaklouta *et al.* 2011] Fatin Zaklouta, Bogdan Stanciulescu et Omar Hamdoun. *Traffic sign classification using kd trees and random forests.* In Neural Networks (IJCNN), The 2011 International Joint Conference on, pages 2151–2155. IEEE, 2011. (Cité en pages 77 et 90.)

Détection de rails, caractérisation de croisements et localisation de trains sur la trajectoire d'un métro automatique

Résumé : Cette thèse porte sur la fonction de détection d'obstacles dans le domaine ferroviaire à partir de la vision par ordinateur. Il s'agit d'assurer une perception de l'environnement situé à l'avant du train afin de détecter et d'évaluer les distances des obstacles situés sur la voie.

Nous avons donc proposé un module détection de rails à partir des images à niveaux de gris, pour déterminer une zone libre d'obstacles à l'avant du train. Cette détection est basée dans l'algorithme de RANSAC et une estimation de la voie par un polynôme de degré 2. Elle s'est montrée robuste à notre base de données et a permis de détecter les rails à des distances supérieures à la distance d'arrêt. Aussi, un algorithme d'étalonnage des caméras installées dans le train a été proposé à partir de la morphologie de la voie.

Comme support de la reconnaissance de rails, nous présentons un module de détection et classification des appareils de voie basé dans le descripteur *Histogram of oriented gradients*(HOG) extrait des images IPM (*Inverse Perspective Mapping*). Un classifieur SVM (*Support Vector Machines*) binaire a été utilisé pour la détection et un SVM multi-classe pour différencier les appareils de voie existants sur la ligne.

Après avoir élaboré le module de détection des rails, nous avons implémenté un détecteur de trains. À partir d'un échantillon d'images de trains de la ligne en question et d'images négatives comme des voitures ou des bus, nous avons créé une base de données d'obstacles pour trouver un descripteur robuste qui arrive à décrire la forme des trains et permet à un classifieur SVM de discriminer les images et détecter les trains. Par la suite, ce classifieur est utilisé par le système global pour déterminer la présence d'un train au-delà de la détection de la voie. À la distance maximale de détection, un rectangle de la taille d'un train est extrait de l'image pour vérifier la présence d'un train. Ces rectangles font l'objet d'une classification au moyen de descripteurs globaux de type HOG et une structure SVM binaire.

Cette étude permettra non seulement de déboucher sur des applications concrètes, mais surtout d'évaluer la maturité des technologies de traitements d'images pour réaliser des fonctions sûres appliquées aux systèmes ferroviaires.

Mots clés : Détection de rails, Détection d'obstacles, Détection d'appareils de voie, Monovision, Étalonnage de la caméra, RANSAC, Trains

Rails detection, turnouts characterisation and trains localization in an automated metro's trajectory

Abstract: This thesis deals with obstacle detection in a railway setting using computer vision. The main task is to provide perception of the environment in front of the train using an optical sensor to detect and evaluate distances to obstacles along the track path.

We present a module for detecting rails from grayscale images to determine an obstacle-free zone in front of the train. This detection is based on the RANSAC algorithm and fitting the track to a second degree polynomial. The method has shown itself robust to our dataset and allows detecting the rails at distances greater than the emergency stopping distance. In addition, a method for calibrating the cameras installed on the train is proposed based on the morphology of the track.

To supplement rail detection, we present a new module for detecting and classifying junctions based on the HOG descriptor extracted from Inverse Perspective Mapping (IPM) images. A Support Vector Machines (SVM) binary classifier was used for detection and a multi-class SVM for distinguishing the junctions along the rails.

In the sequel, a train detector was implemented. Using a set of images of trains found on the studied line and negative images such as cars or buses, we have created a database of obstacles to find a robust descriptor which is able to model the form of trains and allows a SVM classifier to distinguish images and detect trains. Next this classifier is used by the overall system to determine the presence of a train in addition to detecting rails. At the maximum detection distance, a train-sized rectangle is extracted from the image in order to confirm the presence of a train. These rectangles are classified by means of HOG-type global descriptors and a binary SVM structure.

In addition to its applications to concrete problems, this study permits to evaluate the maturity of image processing technologies for fail-safe railway systems.

Keywords: Rail detection, Obstacle detection, Junction detection, Monocular vision, Camera calibration, RANSAC, Trains

MINES ParisTech

ParisTech
INSTITUT DES SCIENCES ET TECHNOLOGIES
PARIS INSTITUTE OF TECHNOLOGY