

**République Algérienne Démocratique et Populaire**  
**Ministère de l'Enseignement Supérieur et de la Recherche**  
**Scientifique**

**Université Batna 2**  
**Faculté des mathématiques et d'informatique**  
**Département d'informatique**

## **Thèse**

*En vue de l'obtention du diplôme de*  
**Doctorat en Informatique**

**Identification des panneaux routiers pour des systèmes**  
**d'aide à la conduite**

*Présentée Par*  
**Yassmina SAADNA**

### **Membres du jury :**

Président de Jury :	Pr. Kamal Eddine MELKEMI	Université Batna 2
Rapporteur :	Dr. Ali BEHLOUL	Université Batna 2
Examineurs :	Dr. Abdelhamid DJEFFAL	Université Biskra
	Dr. Saber BENHARZALLAH	Université Batna 2
	Dr. Larbi GUEZOULI	Université Batna 2

# Dedication

je voudrais remercier toute ma famille et en particulier ma mère et mon père qui m'ont donné l'éducation sans laquelle je n'en serais pas là aujourd'hui. Surtout, je voudrais exprimer mes grands remerciements à mon mari qui m'a aidé et supporté au cours de cette thèse; qui n'a cessé de m'encourager à continuer dans des moments de relâchement et d'hésitation. Mes remerciements vont à mes deux anges Ilyes et Adem pour leur amour continu. Je remercie aussi mes beaux parents pour leur aide précieuse durant mes absences. je remercie mes ami(e)s et tout ceux qui m'ont aidé.

# Remerciements

Avant toute chose, je remercie dieu tout puissant de m'avoir aidé et éclairé le chemin pour la réalisation de cette thèse.

A Monsieur Ali BEHLOUL de m'avoir proposé un tel intéressant sujet, m'ouvrant ainsi les portes sur un domaine de recherche assez vivant. Qui m'a permis, grâce à sa confiance et son soutien précieux, de réaliser ce travail. Pour l'attention qu'il m'a apporté durant toute cette période ainsi que pour sa lecture et ses commentaires qui m'ont permis d'améliorer cette thèse.

A Pr Kamal Eddine MELKEMI d'avoir bien voulu m'honorer en présidant le jury d'examen de cette thèse.

A Dr Abdelhamid DJEFFAL, à Dr Saber BENHARZALLAH et à Dr Larbi GUEZOULI qui ont accepté d'être les examinateurs de ce travail, je les remercie tout particulièrement pour l'attention et le temps qu'ils y ont consacrés.

A Dr Souhila BOUAM pour son soutien et ses conseils.

je remercie tous mes enseignants depuis le primaire jusqu'à l'université.

Je tiens aussi à remercier tous ceux qui ont, de près ou de loin, aidé à rendre ce travail possible, que ce soit par des idées ou par des encouragements.

# Résumé

Les systèmes de détection et de reconnaissance des panneaux routiers fournissent un niveau supplémentaire au sujet de l'assistance au conducteur, ce qui incite la sécurité accrue pour les passagers, les clients de façon quoi de plus les véhicules. Dans le cadre des systèmes avancés d'aide à la conduite, la reconnaissance des panneaux routiers peut être utilisée pour les conducteurs (surtout les personnes ayant des besoins spéciaux) en les alertant sur la présence de panneaux routiers pour réduire les risques de distraction, de fatigue, de mauvaise visibilité et de conditions météorologiques. Bien que plusieurs systèmes de reconnaissance des panneaux routiers aient été proposés dans la littérature, la conception d'un algorithme robuste reste un problème de recherche ouvert. Cette thèse vise à résoudre certains des défis de recherche exceptionnels, tout en tenant compte des variations dans l'éclairage des couleurs, l'échelle, la rotation, la translation, l'occlusion et la complexité de calcul. Le système de reconnaissance des panneaux routiers est composé de trois phases principales : la localisation des panneaux routiers, la détection et la reconnaissance du contenu. Cette thèse présente chaque phase comme un chapitre distinct, dans la première partie plusieurs méthodes de segmentation sont testées et une nouvelle méthode qui améliore la qualité de segmentation est introduite. Dans la deuxième partie, une nouvelle approche basée sur LBP est proposée pour diminuer le nombre des faux positifs ce qui augmente la précision du système. Dans la troisième partie, nous testons plusieurs méthodes de reconnaissance des panneaux. Enfin un système de reconnaissance des panneaux de limitation de vitesse robuste et scalable est développé. Dans la phase de reconnaissance nous utilisons deux classifieurs SVM, le premier pour reconnaître si les régions

détectées sont des panneaux de limitation de vitesse ou non et le deuxième pour classer les chiffres des panneaux. Comme les panneaux de limitation de vitesse ne sont pas toujours clairs et lisibles, à cause de vitesse, manque de lumière ou vibration de la caméra, nous avons proposé d'utiliser un SVM entraîné sur la base MNIST pour reconnaître les chiffres des panneaux de limitation de vitesse. Les objectifs définis pour ce travail ont été entièrement atteints puisque le système développé a atteint un rappel de classification élevé (99,81%) et une haute précision (99,08%) dans un temps de traitement de 11,22 ms sur la base de données GTSDb. L'utilisation de la base de données MNIST pour l'apprentissage rend le système proposé robuste et plus scalable, il donne presque les mêmes résultats sur d'autres bases de données de test comme BTSD où le système obtient un rappel de 99,39% et une précision de 99,05%. Un autre avantage du système proposé est le temps de traitement, où les fausses alarmes détectées dans l'étape de détection sont éliminées avec le premier SVM qui est plus rapide que le second, ainsi le second SVM ne sera pas exécuté seulement si la région détectée est un panneau de limitation de vitesse.

**Mots clés : détection des panneaux routiers; reconnaissance des panneaux routiers; classification; segmentation; reconnaissance des panneaux de limitation de vitesse; SVM; LBP.**

# Abstract

Road sign detection and recognition systems provide an additional level of driver assistance, which enhances the safety of passengers and customers in ways that enhance vehicle safety. As part of an advanced driver assistance systems, the recognition of road signs can be used for drivers (especially those with special needs) by alerting them to the presence of road signs to reduce the risk of distraction. Tiredness, poor visibility and weather conditions. Although several road sign recognition systems have been proposed in the literature; designing a robust algorithm remains an open-ended problem. This thesis aims to solve some of the outstanding research challenges, while taking into account variations in color illumination, scale, rotation, translation, occlusion and computational complexity. The proposed system consists of three main phases : The location of road signs, detection and recognition of content. This thesis presents each phase as a separate chapter, in the first part several segmentation methods are tested and a new method is introduced. In the second part a new approach based on LBP is proposed to reduce the number of false positives which increases the accuracy of the system. In the third part, we test several methods of panel recognition. Finally a robust and scalable speed limit panel recognition system is developed. In the reconnaissance phase we use two SVMs, the first to recognize whether the detected regions are speed limit signs and the second to classify the numbers of the panels. Since the speed limit signs are not always clear and legible, because of speed, lack of light or camera vibration, we proposed to use an SVM trained on the MNIST base to recognize the numbers of speed limitation signs. The objectives defined for this work have been fully achieved since the developed system has achieved a high classification recall

(99.81%) and a high accuracy (99.08%) in a processing time of 11.22 ms on the GTSDDB database. The use of the MNIST database for learning makes the proposed system robust and scalable, it gives almost the same results with other test databases like BTSD where the system gets a 99.39% recall and an accuracy of 99.05%. Another advantage of the proposed system is the processing time, where the false alarms detected in the detection step are eliminated with the first SVM being faster than the second, so the second SVM will not be executed only if the detected region is a speed limit sign.

**keywords : Traffic sign detection; Traffic sign recognition; Classification; Segmentation; Speed limit sign recognition; SVM; LBP**

# ملخص

توفر أنظمة الكشف عن لافتات الطريق والتعرف عليها مستوى إضافي لمساعدة السائق ، مما يعزز سلامة الركاب والعملاء بطرق تعزز سلامة المركبات. كجزء من أنظمة مساعدة السائق المتقدمة ، يمكن استخدام التعرف على إشارات الطريق للسائقين (خاصة أولئك ذوي الاحتياجات الخاصة) عن طريق تنبيههم إلى وجود إشارات الطريق للحد من مخاطر التشتيت. التعب وسوء الرؤية والظروف الجوية. على الرغم من أن العديد من أنظمة التعرف على إشارات الطريق قد تم اقتراحها، إلا أن تصميم خوارزمية قوية لا يزال مشكلة بحثية مفتوحة. تهدف هذه الرسالة إلى حل بعض التحديات البحثية البارزة ، مع الأخذ في الاعتبار التغيرات في إضاءة اللون ، المقياس ، الدوران ، الإغلاق والتعقيد الحسابي. يتكون نظام التعرف على إشارات الطريق من ثلاث مراحل رئيسية: تحديد موقع علامات الطريق في الصورة، تحديد نوعية العلامة والكشف عنها. تقدم هذه الأطروحة كل مرحلة في فصل منفصل، في الجزء الأول يتم اختبار عدة طرق تجزئة للصورة ويتم إدخال طريقة جديدة التي تحسن بدورها من أداء النظام بتقليل عدد الإنذارات الخاطئة. في الجزء الثاني تم اقتراح مقارنة جديدة تعتمد على النمط الثنائي المحلي لتقليل عدد الإيجابيات الكاذبة التي تزيد من دقة النظام. وأخيراً يتم تطوير نظام للتعرف على لوحات حدود السرعة. في مرحلة الاستطلاع ، نستخدم اثنين من شعاع الدعم الآلي ، الأول يتعرف على ما إذا كانت المناطق المكتشفة هي علامات حدود السرعة، و الثاني لتصنيف أرقام اللوحات. نظرًا لأن علامات تحديد السرعة ليست دائمًا واضحة ومقروءة ، بسبب السرعة، نقص الضوء أو اهتزاز الكاميرا ، اقترحنا تدريب شعاع الدعم الأول على قاعدة أرقام مكتوبة بخط اليد للتعرف على أرقام علامات التقييد. وقد تم تحقيق الأهداف المحددة لهذا العمل بشكل كامل حيث أن النظام المقدم قد حقق على قاعدة البيانات الألمانية درجة عالية من التصنيف (99.81%) ودقة عالية (99.08%) في زمن معالجة قدره 11.22 مللي ثانية. استخدام قاعدة بيانات الأرقام المكتوبة بخط اليد للتعلم يجعل النظام المقترح قوي و مرن، وأنه يعطي تقريبا نفس النتائج مع قواعد بيانات اختبار أخرى مثل القاعدة البلجيكية حيث يحصل على درجة عالية من التصنيف 99.39 | % ودقة تعادل 99.05 % . ميزة أخرى للنظام هي وقت المعالجة ، حيث يتم التخلص من الإنذارات الكاذبة التي تم اكتشافها في المصنف الأول ومع كونه سريع ، وبالتالي لن يتم تنفيذ المصنف الثاني إلا إذا كانت المنطقة المكتشفة فعلا علامة الحد الأقصى للسرعة.

**مفاتيح البحث :** الكشف عن إشارات المرور؛ التعرف على إشارات المرور؛ تصنيف البيانات؛ تجزئة الصورة؛ تحديد علامات السرعة القصوى؛ شعاع الدعم الآلي؛ النمط الثنائي المحلي.

# Table des matières

<b>Remerciements</b>	<b>i</b>
<b>Résumé</b>	<b>ii</b>
<b>Table des matières</b>	<b>vii</b>
<b>Liste des figures</b>	<b>ix</b>
<b>Liste des tableaux</b>	<b>xi</b>
<b>Liste des algorithmes</b>	<b>xi</b>
<b>Liste des publications</b>	<b>xiii</b>
<b>1 Introduction générale</b>	<b>1</b>
1.1 Problématique . . . . .	4
1.2 Buts et objectifs de la recherche . . . . .	5
1.3 Les propriétés des panneaux de signalisation routière . . . . .	5
1.4 Structure de la thèse . . . . .	7
<b>2 Etat de l'art</b>	<b>9</b>
2.1 Introduction . . . . .	10
2.2 État de l'art . . . . .	10
2.2.1 La localisation et la détection des panneaux routiers . . . . .	10
2.2.2 Classification . . . . .	29
2.3 Conclusion . . . . .	40
<b>3 Localisation des panneaux routiers</b>	<b>41</b>
3.1 Introduction . . . . .	42
3.2 Background . . . . .	42
3.2.1 Les méthodes de segmentation . . . . .	42
3.2.2 Les espaces de couleur . . . . .	47
3.2.3 Les modèles d'attention visuelle . . . . .	52
3.3 Contribution . . . . .	56
3.4 Conclusion . . . . .	61

<b>4</b>	<b>Détection des panneaux routiers</b>	<b>62</b>
4.1	Introduction . . . . .	63
4.2	Background . . . . .	63
4.2.1	Transformé de hough . . . . .	63
4.2.2	LBP . . . . .	66
4.3	Contribution . . . . .	69
4.4	Conclusion . . . . .	72
<b>5</b>	<b>Reconnaissance des panneaux routiers</b>	<b>73</b>
5.1	Introduction . . . . .	74
5.2	Détection et description des points d'intérêt. . . . .	74
5.2.1	Descripteur SIFT . . . . .	75
5.3	Descripteur SURF . . . . .	78
5.3.1	Appariement . . . . .	80
5.4	Correspondance de modèle (Template matching) . . . . .	82
5.5	SVM (Support Vector Machine) . . . . .	83
5.6	L'histogramme des gradients orientés . . . . .	87
5.7	Conclusion . . . . .	90
<b>6</b>	<b>Système de détection et de reconnaissance des panneaux de limitation de vitesse</b>	<b>91</b>
6.1	Introduction . . . . .	92
6.2	État de l'art . . . . .	92
6.3	Les bases de données disponibles . . . . .	94
6.4	Architecture du système proposé . . . . .	95
6.5	Approche proposée . . . . .	97
6.5.1	Localisation des panneaux de limitation de vitesse . . . . .	97
6.5.2	Détection des panneaux de limitation de vitesse . . . . .	97
6.5.3	Classification des panneaux de limitation de vitesse . . . . .	98
6.6	Résultats expérimentaux . . . . .	101
6.7	Conclusion . . . . .	106
	<b>Conclusion générale</b>	<b>107</b>

# Table des figures

1.1	Les panneaux de danger. . . . .	6
1.2	les panneaux de priorité. . . . .	6
1.3	Les panneaux d'interdiction. . . . .	7
1.4	Les panneaux d'obligation. . . . .	7
1.5	Les panneaux de prescriptions particulières. . . . .	7
2.1	Lookup tables sur la teinte (à gauche) et la saturation (à droite) utilisées dans [1]. . . . .	15
2.2	CDT (colour distance transform) normalisée utilisée par [2] : (a) image originale dans la couleur discrète, (b) CDT noir, (c) CDT blanc, (d) CDT rouge. Les régions ombres signifient une petite distance. . . . .	20
2.3	Dtb (Distance to border) proposé par [3], (a) le blob segmenté, (b) l'image binaire de (a), (c) DtB du blob . . . . .	21
2.4	Illustration de l'approche proposée par [4] (a) un rectangle, (b) un triangle, (c) un cercle, (d) un octogone. . . . .	22
3.1	Le Cube représentant le modèle RGB . . . . .	47
3.2	L'espace de couleur HSI. . . . .	49
3.3	Pyramide HSV : représentation de la couleur en fonction de la teinte, de la saturation et de la luminance. . . . .	50
3.4	L'espace de couleur CIE lab . . . . .	51
3.5	Le modèle d'attention visuel proposé par Itti et al [5]. . . . .	55
3.6	Les images converties en espace HSV, (a) image RGB, (b) image HSV. . . . .	56
3.7	Les images segmentées par la méthode de [6], (a) image RGB, (b) image segmentée. . . . .	57
3.8	Les images segmentées par la méthode de [7], (a) image RGB, (b) image rouge, (c) image bleue, (d) image jaune. . . . .	58
3.9	Les images segmentées par la méthode de [2] (a) et la méthode de [8] (b) . . . . .	58
3.10	Comparaison des images segmentées (a) image RGB, (b) méthode de Lim et al [8], (c) notre méthode proposée. . . . .	60
3.11	Comparaison des cercles détectés dans les images segmentées (a) avec notre méthode proposée, (b) avec la méthode de Lim et al [8]. . . . .	61
4.1	Transformation d'un point dans un cercle. . . . .	66
4.2	Un exemple de LBP basique. . . . .	66
4.3	Exemples de l'opérateur LBP étendu. . . . .	67

4.4	Comparaison des images segmentées (a) image segmentée, (b) image filtrée avec la méthode proposée, (c) image filtrée avec la variance. . . . .	70
4.5	Exemple de la méthode proposée basée sur LBP. . . . .	71
4.6	Effet de la méthode proposée basée sur LBP, (a) sans la méthode proposée, (b) avec la méthode proposée. . . . .	72
5.1	Différents types de points d'intérêt : (a) coin simple, (b) jonction en "V", (c) jonction en "T", (d) jonction en "L", (e) jonction en "damier". . . . .	74
5.2	Détection d'extrema dans l'espace-échelle gaussien. . . . .	76
5.3	Espace d'échelle gaussien. On recherche les extrema des différences de gaussienne afin de déterminer la position des points d'intérêt. . . . .	77
5.4	A gauche, détails des valeurs et directions des gradients; à droite, descripteur SIFT correspondant. Le cercle bleu matérialise la pondération gaussienne suivant l'éloignement du centre de la zone clé [9]. . . . .	78
5.5	Détermination de l'angle de recalage du SURF, en analysant la répartition des réponses des ondelettes de Haar [10]. . . . .	80
5.6	Extraction des différentes composantes du descripteur SURF par le biais des ondelettes de Haar [10]. . . . .	80
5.7	Appariement des images requête avec l'image la plus proche. . . . .	82
5.8	L'hyperplan qui maximise la marge. . . . .	84
5.9	La hiérarchie des divisions pour calculer HOG [11]. . . . .	89
6.1	Architecture du système de reconnaissance des panneaux de limitation de vitesse. . . . .	96
6.2	Exemple des images de la classe positive du premier SVM. . . . .	99
6.3	Exemple des images de la classe négative du premier SVM. . . . .	99
6.4	Exemple des panneaux flous. . . . .	100
6.5	Limitation de vitesse en Europe. . . . .	100
6.6	Exemple de la sous-base MNIST utilisée. . . . .	100

# Liste des tableaux

2.1	Les résultats obtenus par Youssef, A. et al. [12]	19
2.2	Les méthodes basées sur la forme.	23
2.3	Les méthodes de détection avec apprentissage.	28
2.4	Resultats obtenus par Yakimov [13]	30
2.5	Résultats obtenus par Malik, Z. et al [14]	33
2.6	Les méthodes de classification des panneaux routiers avec les caractéristiques extraites manuellement.	34
2.7	Résultats obtenus par Aghdam, H. H. et al[15] comparé avec les résultats de [16]	37
2.8	Les méthodes de classification des panneaux routiers basées sur l'apprentissage profond.	39
6.1	Les bases de données des panneaux routiers.	95
6.2	Comparaison entre la méthode de localisation proposée et la méthode de lim et al. [8].	102
6.3	Comparaison entre LBP et la méthode proposée basée sur LBP.	102
6.4	Résultats de la classification obtenus par le système proposé.	104
6.5	Comparaison du système proposé avec les approches de l'état de l'art.	105

# List of Algorithms

1	Calculer image variance . . . . .	69
2	Reconnaissance des panneaux de limitation de vitesse . . . . .	101

# Liste des publications

- Saadna, Y., Behloul, A. and Mezzoudj, S.(2019).Speed limit sign detection and recognition system using SVM and MNIST datasets.Neural Comput and Applic. <https://doi.org/10.1007/s00521-018-03994-w>.
- Saadna, Y., Behloul, A. (2017). An overview of traffic sign detection and classification methods. International Journal of Multimedia Information Retrieval, 6(3), 193-210.
- Behloul, A., Saadna, Y. (2014). A Fast and Robust Traffic Sign Recognition. International Journal of Innovation and Applied Studies, 5(2), 139.

# **Chapitre 1**

## **Introduction générale**

## Introduction générale

---

Les Nations Unies estiment qu'entre 2010 et 2020, le nombre de morts sur les routes augmentera de 50%, soient environ 1,9 million de personnes. Pour inverser cette tendance, l'ONU a établi, en 2011, la première « Décennie d'action pour la sécurité routière ». Les systèmes d'aide à la conduite peuvent aider à réduire le nombre d'accidents en automatisant des tâches telles que la détection de fatigue, la détection des piétons et la reconnaissance des panneaux routiers. Concevoir des systèmes d'aide à la conduite plus intelligents, visant à minimiser le nombre de mauvaises décisions ou d'accidents, qui peuvent être rencontrés pendant le trajet, est l'un des sujets d'actualité de la technologie automobile d'aujourd'hui et de plus en plus un grand nombre de personnes s'équipent aujourd'hui d'un assistant d'aide à la conduite. Les systèmes d'aide à la conduite assistent le conducteur et permettant d'éviter de graves erreurs et parfois même de les corriger, ils permettent aux conducteurs de conduire sans stress.

La reconnaissance des panneaux de signalisation a reçu un intérêt croissant ces dernières années, il est même considéré comme une caractéristique très importante des véhicules intelligents. Les panneaux de signalisation portent des informations utiles substantielles qui pourraient être ignorées par les conducteurs en raison de la fatigue ou de la recherche d'une adresse, etc. Ces conducteurs sont également susceptibles de porter moins d'attention aux panneaux de signalisation lors de la conduite dans des conditions météorologiques menaçantes. Par conséquent, la détection et la reconnaissance automatique des panneaux routiers devient indispensables pour réduire le nombre de victimes sur la route.

Un système de reconnaissance des panneaux de signalisation est généralement divisé en trois parties : la localisation, la détection et la reconnaissance. Les panneaux de signalisation ont des couleurs et des formes spécifiques qui peuvent être facilement observées par les conducteurs. La détection consiste à extraire le panneau de signalisation d'une scène en plein air, elle repose généralement sur les caractéristiques inhérentes aux panneaux de signalisation (tels que la couleur et la forme). La majorité des travaux de détection des panneaux routiers sont initiés à l'aide des informations de couleur des panneaux routiers appelés segmentation des couleurs. La segmentation des couleurs est plus fréquemment effectuée en utilisant un espace colorimétrique. Les espaces colorimétriques les plus utilisés à cette fin sont HSI, RGB, CIElab, YCbCr et HSV. Dans les approches de segmentation à base de couleurs, les images sont d'abord converties en un espace de couleur désigné, puis un algorithme de segmentation, par ex. seuillage, est appliqué. Plusieurs auteurs préfèrent d'utiliser l'espace RGB car les conversions entre les espaces de couleurs sont non linéaires, un simple seuillage est appliqué directement dans l'espace RGB par [17], [18], [19], [20], [7]. Pour éviter le problème du choix de seuil, plusieurs auteurs utilisent un filtre comme dans [2], [8], [21]. Comme l'espace RGB est sensible aux changements de lumière, plusieurs auteurs préfèrent d'utiliser l'espace HSV car ils supposent que la teinte H est invariante aux changements de luminance. Les auteurs dans [1], [22] utilisent un seuillage dans le canal Hue afin de déterminer la couleur dominante. Fleyeh.H [23] a utilisé un seuillage dans les composantes H, S et V, cette méthode résiste aux changements d'éclairage mais elle est coûteuse en temps de calcul. Une agrégation dynamique des pixels est utilisée dans

[24] pour segmenter l'image. L'algorithme de Otsu est utilisé par [25, 26, 27] pour trouver le seuil optimal dans l'espace HSV. Shar, a et al. [28] et Escalera et al. [1] ont choisi l'espace HSI car ils supposent que la composante Hue de l'espace HSI est invariante aux changements de luminance. Dans [29] et [30] l'espace YUV est choisi.

En conjonction avec les approches basées sur les couleurs, les approches basées sur les formes sont rarement utilisées dans la détection des panneaux routiers. Dans ce cas, les informations de couleur sont soit utilisées en pré-traitement, soit jamais utilisées du tout. Hechri et al. [31] utilisent le template matching, Garcia-Garrido et al. [32] et Miura et al. [29] utilisent la transformée de Hough pour détecter les panneaux routiers. la Transformation de Symétrie Radiale (TSR) est utilisée par [33] et [34]. Youssef, A. et al. [12] ont choisi la transformé de HOG (Histogram of Oriented Gradient).

Les deux méthodes précédentes (colorimétrique et géométrique) ne sont pas robustes aux changements d'éclairage, changements d'échelle, les occlusions, les rotations, etc. Cependant les méthodes d'apprentissage peuvent pallier ce problème. Viola et Jones [35] utilisent l'algorithme de AdaBoost basé sur les ondelettes de Haar, après, les auteurs dans [36, 37] remarquent que le détecteur de Viola et Jones est robuste aux bruits mais il génère beaucoup de faux positifs. Chen, T. et al. [38] et Priscariu et al. [39] exploitent à la fois le Adaboost et le SVM afin de détecter les panneaux. Les algorithmes génétiques sont utilisés par Escalera et al. [1] et Fang et al. [40] utilisent les réseaux de neurones. Aghda, et al. [15] proposent un ConvNet léger et optimisé avec une fenêtre coulissante pour détecter les panneaux routiers, Wu et al [41] de leurs part utilisent le CNN pour localiser les panneaux routiers, Zang et al. [42] combine le CNN avec le détecteur de LBP. Les méthodes basées sur CNN peuvent obtenir une grande précision de détection, mais elles sont plus complexes et le temps de formation est trop long et la limitation doit encore être améliorée.

Dans la phase de reconnaissance, les auteurs utilisent soit les méthodes de deep learning, ou bien les méthodes d'apprentissage classiques qui utilisent des descripteurs comme : l'histogramme des gradients orientés (HOG) utilisé par [43, 44], LBP [45, 46], le descripteur SIFT [47, 48, 49, 50], le descripteur SURF [4, 46, 51]. L'un des principaux problèmes de cette approche est que la précision de la reconnaissance est largement déterminée par la capacité du descripteur à représenter les caractéristiques discriminantes, pour pallier ce problème plusieurs auteurs passent au deep learning [15, 52, 53, 54, 55, 16, 52, 56, 57], cependant, les architectures de deep learning nécessitent une grande base de données annotée et prennent un temps d'apprentissage important.

Dans le monde réel, l'efficacité de la détection et de la reconnaissance des panneaux de signalisation dépend de nombreux facteurs tels que les variations d'éclairage, les changements des conditions météorologiques, les occlusions par d'autres objets, les rotations des panneaux capturés lors des virages et le changement de taille des panneaux, qui peuvent éventuellement diminuer les performances du système de reconnaissance des panneaux.

### 1.1 Problématique

Le problème principal des systèmes de reconnaissance des panneaux routiers n'est pas de détecter ou de classifier le maximum des panneaux, mais également d'obtenir une grande précision dans une grande bases de données. Il s'agit d'un système multi-objectif, détecter et classifier tous les panneaux en temps réel et diminuer le nombre des fausses alarmes qui peuvent gêner le conducteur comme nous avons illustré dans l'exemple [58]. Un système de reconnaissance des panneaux routiers installé dans un smartphone avec une caméra de 30 frames par seconde, (108 000 images dans une vidéo d'une heure), si nous supposons que chaque 4 minutes nous détectons un panneau qui s'étend sur 2 secondes (selon la vitesse de véhicule), cela signifie que au bout de 1h, nous trouverons au total 15 panneaux et que chacun affichera dans 60 frames (900 frames qui contiennent des panneau alors que 107 100 ne le font pas). En supposant également que le système présente un taux de faux positifs de 1%, cela signifie qu'il y a 17 fausses alarmes détectées en 1 minute (1071 en 1h), dans 4 minutes 1 vrais positifs et 68 fausses alarmes. Cela pourrait éventuellement amener de nombreux utilisateurs à désactiver cette application.

Comme la reconnaissance des panneaux routiers est considéré comme un problème de détection et de reconnaissance d'objet dans des scènes complexes qui peuvent contenir quelques problèmes tels que :

- Le changement d'éclairage, selon le moment de la journée, la saison, la nébulosité et d'autres conditions météorologiques;
- La couleur s'estompe : en raison de la longue exposition au soleil et à la pluie, les signes auront un effet de fondu et ne seront pas clairs.
- Changement d'échelle, car la taille du panneau dépend de la distance à la caméra;
- Rotation et translations, car les panneaux peuvent être capturés selon différents points de vue;
- La présence d'obstacles dans la scène pouvant occulter partiellement ou totalement le panneau;
- L'existence d'un certain nombre d'objets similaires soit en couleur ou en forme;
- La condition physique des panneaux qui se changent avec le temps;
- Similarité et normalisation des panneaux : différents pays ont des ensembles des panneaux normalisés différents et il existe certaines similitudes entre les différentes catégories des panneaux.
- Mouvement du véhicule : En raison des images de signe prises par la caméra embarquée, le mouvement du véhicule entraîne un tremblement de la caméra et rend l'image floue.
- La quantité d'informations dans l'image est grande et ça prend du temps pour traiter une telle image et détecter l'objet désiré.

### 1.2 Buts et objectifs de la recherche

Le but de cette thèse est de développer des algorithmes de détection et de classification robustes pour les panneaux de signalisation routière, donnant naissance à des systèmes capables de renforcer la nature irréprochable de l'ADAS (Advanced Driver Assistance System). Cette thèse vise à résoudre certains des défis de recherche exceptionnelles en TSR (Traffic Sign Recognition), tout en tenant compte des variations dans l'éclairage des couleurs, l'échelle, la rotation, la translation, l'occlusion, la complexité de calcul et les limitations fonctionnelles.

L'objectif global est de développer un système de reconnaissance des panneaux routiers robuste, précis, simple et rapide. Les objectifs spécifiques de cette recherche sont énumérés ci-dessous :

- Proposer une nouvelle méthode simple, rapide et efficace pour localiser les panneaux routiers;
- Proposer une nouvelle méthode basée sur LBP pour filtrer les images segmentées afin de réduire le nombre des fausses alarmes;
- Utiliser une stratégie de classification des panneaux simple et efficace;
- Développer un système de reconnaissances des panneaux de limitation de vitesse robuste, scalable et rapide;

### 1.3 Les propriétés des panneaux de signalisation routière

Avant d'étudier les méthodes de détection et de classification des panneaux de signalisation routière, il est utile de passer en revue les données sur les classes des panneaux et leurs caractéristiques. Les panneaux routiers et les feux de signalisation et autres dispositifs de circulation sont utilisés pour avertir, guider ou informer les usagers de la route. Ils aident à atteindre un niveau acceptable de qualité du trafic routier et à accroître la sécurité grâce à un mouvement ordonné et prévisible de tout le trafic, à la fois pour les véhicules et les piétons.

Les panneaux routiers sont conçus pour être facilement reconnus par les conducteurs, principalement parce que leurs formes et leurs couleurs se distinguent facilement de leur environnement. Ces caractéristiques discriminantes permettent de classer les panneaux en plusieurs classes, selon leurs formes et leurs couleurs.

La Convention de Vienne sur la signalisation routière, signée le 8 novembre 1968, reconnaît que l'uniformité internationale des panneaux routiers est nécessaire pour faciliter la circulation routière et pour augmenter la sécurité sur la route. Les panneaux sont classifiés en sept catégories :

## Introduction générale

---

- Les panneaux de danger : est un type de signalisation routière qui indique un danger sur la route. Il se caractérise par un triangle équilatéral avec un bord rouge épais et un intérieur blanc ou jaune. Un pictogramme dedans sert à spécifier différents avertissements. La figure 1.1 illustre quelques exemples des panneaux de danger.



FIGURE 1.1 – Les panneaux de danger.

- Les panneaux de priorité : ils sont destinés à notifier ou à porter à la connaissance des usagers de la route des règles particulières de priorité à des intersections comme est illustré dans la figure 1.2.

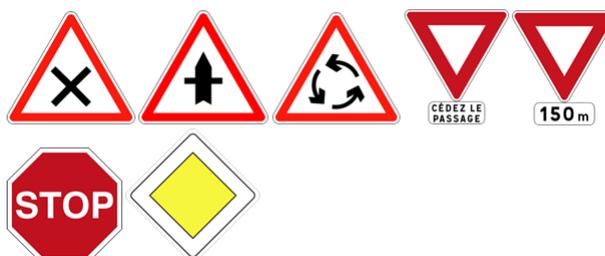


FIGURE 1.2 – les panneaux de priorité.

- Les panneaux d'interdiction ou de restriction : est un type de signalisation routière qui signale à l'utilisateur de la route qu'il doit respecter l'interdiction indiquée. Ils sont circulaires avec un fond blanc, jaune ou bleu, bordé d'une couronne rouge et comme est illustré dans la figure 1.3.
- Les panneaux d'obligation : sont des panneaux de positions, placés à l'endroit même où ils obligent à effectuer une action. Ils peuvent aussi indiquer des voies réservées à certains usagers. Ils se caractérisent par une forme circulaire, un fond bleu et un pictogramme blanc. La figure 1.4 montre quelques exemples.
- Les panneaux de prescriptions particulières : sont généralement carrés ou circulaires, à fond bleu avec un symbole ou une inscription de couleur claire, ou à fond clair avec un symbole ou une inscription de couleur foncée. La figure 1.5 illustre quelques exemples de panneaux de prescriptions particulières.



tion). Pour la compréhension du lecteur, les approches que les chercheurs ont adoptées au fil des années sont classifiées et tabulées. La revue de la littérature est effectuée pour les approches clés qui sont spécifiquement utilisées dans la détection et la reconnaissance des panneaux de signalisation.

**Chapitre 3** : résume brièvement les théories de base sur lesquelles les nouveaux algorithmes de localisation des panneaux routiers ont été proposés. Le chapitre discute les méthodes de segmentation, les espaces de couleur et les modèles d'attention visuelle. La section 3.3 présente une comparaison entre les méthodes de segmentation des espaces HSV et RGB, ainsi notre contribution dans la localisation des panneaux routiers où nous proposons une nouvelle méthode de segmentation qui augmente la précision du système.

**Chapitre 4** : présente brièvement les méthodes utilisées dans l'étape de détection. La section 4.3 détaille notre contribution pour la détection des panneaux routiers, une nouvelle méthode basée sur LBP est proposée pour filtrer l'image afin de réduire les fausses alarmes.

**Chapitre 5** : résume les méthodes de reconnaissance des panneaux routiers utilisés dans cette thèse. Le chapitre couvre les détails des algorithmes de reconnaissance utilisés pour identifier les panneaux de signalisation dans la scène. Nous comparons entre plusieurs méthodes : template matching, les points d'intérêt et le SVM.

**Chapitre 6** : présente le développement d'un système de reconnaissance des panneaux de limitation de vitesse en utilisons les approches proposées dans les chapitres précédents pour la détection des panneaux. Ensuite, dans la phase de classification nous proposons d'utiliser le SVM entraîné sur la base GTSRB et MNIST pour la classification et la reconnaissance des chiffres des panneaux. Le système proposé est testé dans différentes conditions météorologiques et conditions de luminosité. Pour tester la scalabilité le système est testé sur deux autre base de données BTSD et STS.

**Conclusion générale** : Ce chapitre résume les principales contributions et conclusions tirées de cette recherche. Il présente les plans futurs qui peuvent être mis en œuvre pour améliorer le travail et les questions à approfondir.

## **Chapitre 2**

### **Etat de l'art**

### 2.1 Introduction

Le domaine de la détection et de la reconnaissance des panneaux routiers a reçu un intérêt croissant ces dernières années en raison de leurs nature difficile en tant que problème majeur dans l'analyse des scènes routières dans le traitement d'image. Au cours des dernières années, un certain nombre de systèmes d'assistance aux conducteurs ont été proposés et mis en oeuvre, et de nombreuses recherches ont été réalisées au sujet de la reconnaissance des panneaux routiers, la plupart adoptent une approche en trois phases : la localisation des régions d'intérêt qui contiennent les panneaux, la détection des panneaux et l'identification de leurs formes géométriques, la troisième phase permet de reconnaître le type de chaque panneau. Certains auteurs fusionnent les deux premières phases car ils n'utilisent pas la couleur pour localiser les panneaux donc ils optent pour un système en deux phases : détection et classification. La phase de localisation et détection est très importante, car elle sert à détecter tous les panneaux et éliminer au maximum les autres objets qui peuvent apparaître dans l'image, cela permet de diminuer le taux des faux positifs. Dans le cas d'une fausse alarme dans la phase de détection, les performances seront plus faibles dans l'étape de classification, car le classifieur n'est pas toujours entraîné sur les faux positifs.

Dans ce chapitre nous allons présenter un aperçu sur les méthodes de détection et de classification récentes et efficaces. Nous avons classifié les méthodes de détection en trois catégories : les méthodes colorimétriques qui utilisent la segmentation pour localiser les panneaux, les méthodes géométriques et les méthodes avec apprentissage. Les méthodes de classification sont classifiées en deux classes les méthodes qui utilisent des caractéristiques extraits par un expert et les méthodes de l'apprentissage profond (deep learning).

### 2.2 État de l'art

Les systèmes de reconnaissance des panneaux routiers se composent de trois phases principales : la localisation qui sert à localiser les régions d'intérêts qui peuvent contenir des panneaux routiers, la détection des panneaux et la reconnaissance de leurs formes géométriques, la troisième étape consiste à classifier le panneau détecté pour reconnaître la signification de son pictogramme.

#### 2.2.1 La localisation et la détection des panneaux routiers

Les méthodes de détections peuvent être classifiés en trois classes principales : les méthodes colorimétriques qui utilisent la segmentation pour localiser les régions d'intérêts, les méthodes géométriques et les méthodes par apprentissage.

### 2.2.1.1 Les méthodes colorimétriques

Une segmentation à base de couleur dominante est appliquée afin de détecter des régions d'intérêts, ces régions sont ensuite filtrées selon un algorithme de reconnaissance ou un modèle d'apparence. Ces méthodes sont rapides, mais leur inconvénient majeur est qu'elles sont sensibles aux changements d'éclairage. Les auteurs travaillent sur différents espaces de couleur afin de pallier ce type de problème, nous classifions les méthodes de segmentation selon les espaces de couleurs utilisés.

#### (A) Les méthodes qui utilisent l'espace RGB

La raison d'utiliser l'espace RGB est d'économiser le temps de calcul des conversions entre les espaces de couleur. Pour éviter les effets des variations de luminosité suivant l'heure, la saison ou les conditions météorologiques, il faut fixer des seuils  $\alpha$ ,  $\beta$ , et  $\gamma$ . Le plus souvent, ces seuils sont fixés après des expérimentations sur un échantillon d'images.

Soheilian et al.[17] ont testé deux méthodes de détection de couleurs qui sont basées sur les espaces CIE Lab et HSV dans le but de pallier le problème de variation de luminance, mais ils ont observé que dans un changement important d'éclairage, les résultats obtenus ne sont pas satisfaisants, car les composantes chromatiques ne sont pas complètement indépendantes de la quantité de lumière. C'est pour cela ils ont choisi de travailler avec les ratios de G et B sur R selon une méthode de classification des pixels présentée dans [59] :

$$I(x, y) = \begin{cases} C_R & \text{si } \frac{R}{B} > T \quad \text{et} \quad \frac{R}{G} > T \\ C_B & \text{si } \frac{B}{R} > T \quad \text{et} \quad \frac{B}{G} > T \end{cases} \quad (2.1)$$

Où :

$I(x, y)$  : représente le pixel de coordonnées  $x, y$  de l'image  $I$ .

$R, G, B$  : les valeurs des trois canaux rouge, vert et bleu du pixel  $I(x, y)$ .

$C_B, C_R$  : classes rouge et bleue.

Shneier et al. [18] de sa part utilise le rapport entre les canaux :

$$\frac{R}{G} > \alpha \quad \text{Et} \quad \frac{R}{B} > \beta \quad \text{Et} \quad \frac{G}{B} > \gamma \quad (2.2)$$

Torresen et al.[19] ainsi que Benallal et al. [20] utilisent aussi un seuillage pour segmenter l'image :

$$R > \alpha \quad \text{Et} \quad (R - G) > \beta \quad \text{Et} \quad (R - B) > \gamma \quad (2.3)$$

L'inconvénient de ces méthodes est la difficulté de fixer les seuils  $T, \alpha, \beta$  et  $\gamma$ . Ainsi que les panneaux de signalisation temporaire (les panneaux jaunes) ne sont pas traités.

Pour définir la couleur rouge, les auteurs dans [21] ont appliqué une condition sur les trois canaux  $2R > G + B$  afin de pallier le problème des seuils. Une autre approche est proposée par [60], il a défini une région cylindrique dans l'espace RGB pour segmenter l'image.

D'autres auteurs ont travaillé aussi sur l'espace RGB comme [61, 62, 63, 2, 64] mais ces méthodes sont fortement liées aux seuils choisis, ce qui rend la comparaison de leurs performances un processus ardu.

Un filtrage est appliqué dans [2] pour chaque pixel  $X = [R, G, B]$  et  $S = R + G + B$  :

$$f_R(x) = \max(0, \min(R - G, R - B) / S) \quad (2.4)$$

$$f_B(x) = \max(0, \min(B - R, B - G) / S) \quad (2.5)$$

$$f_Y(x) = \max(0, \min(R - B, G - B) / S) \quad (2.6)$$

Les deux premières équations 2.4 et 2.5 extraient les fragments de l'image rouge ou bleu respectivement, comme la couleur d'un jaune pur a la même valeur dans les canaux rouge et vert, et un zéro dans le canal bleu. La troisième équation 2.6 extrait les fragments jaunes. Alors pour chaque image RGB ils associent trois images rouge, bleu et jaune, et la couleur dominante possède une forte intensité, mais les signaux détériorés ont des faibles intensités.

Dans une autre approche de [7], ils ont basé sur SVF (Simple Vector Filter) proposé par [65], ils ont trouvé dans leurs expériences que cette approche donne de bons résultats dans la séparation des éléments de la couleur rouge, bleu, jaune. Pour chaque pixel  $x = [X_R, X_G, X_B]$  le SVF sera calculé selon les équations suivantes :

$$\text{Rouge} : R - G > 40 \quad \text{et} \quad R - B > 40; \quad (2.7)$$

$$\text{Jaune} : R - B > 50 \quad \text{et} \quad G > 50; \quad (2.8)$$

$$\text{Bleu} : B - R > 65 \quad \text{et} \quad B > 45. \quad (2.9)$$

Ils ont séparé leur base d'images en deux parties "ensoleillé" et "nuageux", selon les changements d'illumination. Dans la première ils ont détecté 2113 panneaux parmi 2173 avec 269 de faux positifs. Dans la deuxième, 2562 panneaux sont détectés parmi 2614 avec 299 faux positifs. Cette phase de segmentation prend 60 ms en moyen.

Lim et al.[8] de leur part ont préféré l'espace R'G'B', d'abord ils normalisent les trois canaux RGB par l'intensité I :

$$I = \frac{R + G + B}{3} \quad (2.10)$$

$$r = \frac{R}{I}, \quad g = \frac{G}{I}, \quad b = \frac{B}{I} \quad (2.11)$$

Après, ils construisent quatre nouvelles images selon l'équation donnée dans[5] :

$$R' = r - \frac{(g + b)}{2} \quad (2.12)$$

$$G' = g - \frac{(r + b)}{2} \quad (2.13)$$

$$B' = b - \frac{(r + g)}{2} \quad (2.14)$$

$$Y' = \frac{r + g}{2} - \frac{|r - g|}{2} - b \quad (2.15)$$

La couleur dominante aura une grande intensité ce qui facilite l'extraction des panneaux, ensuite ils appliquent un seuillage (thresholding) pour binariser les quatre images (R, G, B, Y) et appliquent des opérations morphologiques pour éliminer les pixels indésirables. Leur approche est capable de détecter 93.63% des panneaux.

Pour éviter le problème de seuillage, les auteurs dans [66] et [67] ont préféré d'utiliser les réseaux de neurones et le SVM pour classifier les couleurs.

**(B) Les méthodes qui utilisent l'espace HSV**

Les transformations de coordonnées RGB en HSV et réciproquement ne sont pas linéaire, mais les auteurs préfèrent de l'utiliser car ils supposent que la teinte H est invariante aux changements de luminance comme [1], [22], ils utilisent un seuillage dans le canal Hue afin de déterminer la couleur dominante.

Wu et al.[26] ont transformé l'image RGB en HSV suivant la formule suivante :

$$H = \begin{cases} H & \text{si } B \leq G \\ 360 - H & \text{sinon} \end{cases} \quad (2.16)$$

$$S = \frac{\max(R, G, B) - \min(R, G, B)}{\max(R, G, B)} \quad (2.17)$$

$$V = \frac{\max(R, G, B)}{255} \quad (2.18)$$

Tel que :

$$H = \cos^{-1} \left( \frac{0.5[(R - G) + (R - B)]}{\sqrt{(R - G)^2 + (R - B)(G - B)}} \right) \quad (2.19)$$

Maldonado et al.[63] fixent un seuil sur la composante H et S, combiné avec la décomposition achromatique, simple et rapide mais il n'est pas robuste aux changements d'illumination et les détériorations des signaux, Fleyeh.H [23] a utilisé un seuillage dans les composantes H, S et V, cette méthode résiste aux changements d'éclairage mais elle est coûteuse en temps de calcul. [24] utilisent une tech-

nique d'agrégation dynamique des pixels à partir du pixel source pour segmenter l'image.

Les auteurs dans [25, 26, 27] ont utilisé l'algorithme de Otsu [68] pour trouver les seuils dans l'espace HSV, et dans [27] ils se sont basés sur la logique floue pour localiser les régions d'intérêt.

Mais l'espace HSV n'est pas uniforme, c'est-à-dire qu'une distance entre couleurs visuellement proches calculée dans cet espace peut être très grande.

Yakimov et al. [6] a considéré qu'il n'est pas possible de détecter les panneaux de circulation dans les images fixes en appliquant directement un simple seuillage dans l'espace RGB en raison des variations d'éclairage, ce qui les a incités à choisir l'espace HSV. Ils ont utilisé la méthode expérimentale pour limiter les valeurs des seuils optimales pour la couleur rouge comme est illustré dans l'équation suivante :

$$\begin{aligned} (0.0 \leq H < 23) \parallel (350 < H < 360) \\ (0.85 < S \leq 1) \\ (0.85 < V \leq 1) \end{aligned} \tag{2.20}$$

Donc le pixel est considéré comme rouge si les composante H, S et V respectent les conditions de l'équation 2.20. Après la segmentation ils ont utilisé un algorithme présenté dans [69] pour éliminer le bruit des images segmentées. L'avantage cet algorithme est que seul le bruit sera supprimé, cependant, les régions d'intérêt restent non filtrées.

Wang et al.[70] ont également choisi l'espace HSV, ils ont trouvé que la méthode de seuillage classique donne de bons résultats dans de nombreuses différentes conditions d'éclairage différentes, sauf dans les cas de fonte de couleur ou de mauvaises conditions d'éclairage. Ils ont proposé une nouvelle méthode de seuillage en utilisant les informations de couleur des pixels voisins. Premièrement, le degré rouge de chaque point  $x$  est calculé pour obtenir une nouvelle image  $f_R(x)$  avec l'équation suivante :

$$f_R(x) = \begin{cases} S(x) \frac{\sin(H(x)-300)}{\sin(60)} & \text{si } H(x) \in [300, 360] \\ S(x) \frac{\sin(60)-H(x)}{\sin(60)} & \text{si } H(x) \in [0, 60] \\ 0 & \text{Autres} \end{cases} \tag{2.21}$$

Deuxièmement, le degré rouge normalisé  $f_{NR}(x)$  est calculé comme suit :

$$f_{NR}(x) = \frac{(f_R(x) - \mu_R(\omega_x))}{(\sigma_R(\omega_x))} \tag{2.22}$$

$\mu_R(\omega_x)$  et  $\sigma_R(\omega_x)$  sont respectivement la moyenne et la variance des degrés rouges des pixels dans la fenêtre  $\omega_x$  centrée sur  $x$ .

Troisièmement, l'intensité normalisée  $f_{NI}$  est calculée avec l'équation suivante :

$$f_{NI}(x) = \frac{(f_I(x) - \mu_I(\omega_x))}{(\sigma_I(\omega_x))} \quad (2.23)$$

$f_{NI}(x)$  est l'intensité du pixel  $x$ ,  $\mu_I(\omega_x)$  et  $\sigma_I(\omega_x)$  sont respectivement la moyenne et la variance des intensités des pixels dans la fenêtre  $\omega_x$ . Enfin le rouge R est donné comme suit :

$$R(x) = \begin{cases} 1 & \text{si } f_{NR}(x) > \max(THR1, f_{NI} + THR2) \\ 0 & \text{Autres} \end{cases} \quad (2.24)$$

$THR1$  et  $THR2$  sont des seuils fixés expérimentalement.

(C) **Les méthodes qui utilisent l'espace HSI**

Shar,a et al.[28] ont supposé que la composante Hue de l'espace HSI est invariante aux changements de luminance, c'est ce qui a poussé les auteurs à choisir cet espace de couleur.

Escalera et al.[1] ont choisi l'espace HSI pour détecter les panneaux, ils ont pris en considération que les composantes H et S afin de pallier le problème de variations de luminosité. Ils ont construit deux look-up-tables (LUTs) comme est illustré dans la figure 2.1, une pour la composante H et une autre pour la composante S. l'idée est que chaque LUT compense l'autre, si une composante a des fausses valeurs l'autre peut l'aider.

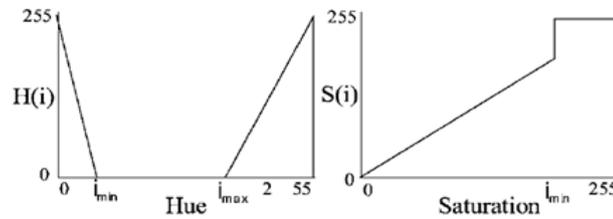


FIGURE 2.1 – Lookup tables sur la teinte (à gauche) et la saturation (à droite) utilisées dans [1].

$H(i)$  et  $S(j)$  sont calculé comme suit :

$$H(i) = \begin{cases} 255 * \frac{i_{min}-i}{i_{min}} & 0 < i < i_{min} \\ 0 & i_{min} < i < i_{max} \\ 255 * \frac{i_{max}-i}{i_{max}} & i_{max} < i < 255 \end{cases} \quad (2.25)$$

$i$  : L'ancienne valeur de la composante Hue,  $H(i)$  est la nouvelle valeur.

$$S(j) = \begin{cases} j & 0 < j < j_{min} \\ 255 & j_{min} < j < 255 \end{cases} \quad (2.26)$$

$j$  : L'ancienne valeur de la composante Saturation,  $S(j)$  est la nouvelle valeur.

Une fois les LUTs sont calculées, ils multiplient ensuite les images obtenues par rapport au ET logique.

Et même dans [71, 72, 73] le choix s'est porté sur l'espace HSI, ces méthodes sont robustes aux changements d'éclairage mais elles consomment beaucoup de temps dans le passage de l'espace RGB à HSI.

Fang et al. [74] supposent de leur côté que chaque couleur particulière d'un signal peut être représentée par une valeur de teinte distribuée de manière gaussienne avec une variance  $\delta^2$ . L'ensemble de toutes ces valeurs de teinte est noté  $[h_1, h_2, \dots, h_q]$ . Les auteurs calculent un degré de similarité  $z$  entre la teinte  $h$  d'un pixel et les teintes  $h_k$  des couleurs de signaux comme suit :

$$z = \max_{k=1, \dots, q} (Z_k) \quad \text{tel que} \quad Z_k = \frac{1}{\sqrt{2\pi}\delta} \exp(-(h - h_k)^2 / 2\delta^2) \quad (2.27)$$

Ainsi, ils obtiennent, non pas une image segmentée, mais une image où chaque valeur de pixel représente la similarité entre la couleur du pixel et les couleurs standard des pixels de signaux [75], mais les calculs ne sont pas linéaires.

### (D) Les méthodes qui utilisent l'espace YUV

Les panneaux rectangulaires d'information étant détectés dans [29] par un seuillage colorimétrique dans YUV suivi d'une projection horizontale et verticale du gradient et d'une reconnaissance du Kanji (système d'écriture japonaise). Les performances de cette approche sont seulement illustrées par quelques exemples. Et même dans [30] le choix s'est porté sur l'espace YUV après une correction chromatique préalable compte tenu des valeurs des pixels de la chaussée théoriquement gris ( $R=G=B$ ) [76].

Dans [77], les auteurs ont comparé différentes méthodes de segmentation afin de trouver la meilleure méthode dans le domaine de la reconnaissance des panneaux routiers. Ils ont classifié les méthodes de segmentation en trois catégories principales : la segmentation avec binarisation, la décomposition chromatique / achromatique et la détection des contours. Ils ont ensuite proposé une nouvelle méthode de segmentation dans laquelle ils combinaient SVM avec LUT (Look Up Table). Après avoir mis en œuvre les différentes méthodes et mené une recherche approfondie pour trouver la meilleure méthode, ils ont conclu que :

- Dans les images individuelles, les meilleurs résultats ont été obtenus avec la méthode RGB; cependant, dans les vidéos, les meilleurs résultats ont été observés lors de l'application de la méthode LUT SVM;
- La détection de contours peut être utilisée en complément de la méthode de segmentation mais ne peut pas être utilisée seule.
- La standardisation de l'espace colorimétrique RGB prouve une bonne performance avec moins d'opérations. De l'autre côté, l'espace HSV, bien qu'il donne des résultats plus élevés, prend beaucoup de temps d'exécution, ce qui en fait

une méthode moins efficace. Ainsi, pourquoi utiliserions-nous une transformation non linéaire si la simple normalisation est suffisante ?

Donc la transformation non linéaire entre les espaces de couleurs n'est pas nécessaire tant que une simple normalisation est suffisante pour obtenir les mêmes résultats.

Contrairement aux méthodes de seuillage de couleur et d'extraction des régions extrêmes utilisées dans les approches précédentes, une approche récente proposé par [78], ils ont utilisé "High-Contrast Region Extraction (HCRE)", motivée par les méthodes de détection en cascade, pour extraire la région d'intérêt avec un contraste local élevé. Cela permet de garder un compromis entre les taux de détection et d'extraction. Profitant de l'observation selon laquelle différents types de panneaux ont un contraste relativement élevé dans les régions locales. Le HCRE peut rejeter environ 83,10% des régions non intéressantes à faible contraste local comme le ciel, les routes et certains bâtiments, il augmentant ainsi la vitesse de détection du détecteur SFC-tree proposée par [79] de 5 à plus de 10 images par seconde dans leurs expériences.

### 2.2.1.2 Les méthode géométriques

Dans le cas d'une telle approche les auteurs ne considèrent pas la segmentation couleur absolument fiables en raison de sa sensibilité à divers facteurs, tels que la distance de la cible, les conditions météorologiques, le temps de la journée, ou la réflexion des signes, la détection des panneaux se fait à partir des contours de l'image analysée par une approche structurelle ou globale. Ces méthodes sont généralement plus robustes que celles photométriques parce qu'elles traitent le gradient de l'image et peuvent aussi traiter des images au niveaux de gris [76], mais elles sont coûteuses en temps de calcul, car la cadence du traitement dépend du nombre de contours détectés. Cependant, les méthodes géométriques peuvent traiter les images au niveau de gris, mais dans certains pays, comme le Japon, il y a des paires de signes au niveau de gris apparaissent exactement les mêmes. Pour pouvoir les distinguer, la couleur est absolument nécessaire [2]. D'autre part, certains auteurs utilisent l'une des méthodes calorimétriques pour localiser la région d'intérêt, ensuite compléter avec des méthodes géométriques afin de détecter la position des panneaux et de reconnaître sa forme géométrique.

Dans une autre approche de Vitabile et al. [24] chaque forme de signal routier est représentée par une image binaire de taille fixe (36\*36 pixels). Et après détection des régions d'intérêt avec la technique d'agrégation dynamique des pixels, ils les remettent à la même échelle des images binaires représentant les formes, afin de calculer une mesure de similarité avec toutes les formes des panneaux routiers en utilisant le coefficient de Tanimoto : si X et Y sont les ensembles de pixels des images binaires comparées [75], le coefficient de Tanimoto S est défini comme suit :

$$S = \frac{|x \cap y|}{|x \cup y|} \quad (2.28)$$

La valeur de  $S$  est normalisée, plus elle est proche de 1 plus le modèle et la région d'intérêt sont similaires. Les auteurs obtiennent plus de 86% de détections parmi 620 images de 24 classes.

Template matching est utilisé par Hechri et al. [31] pour filtrer les régions qui ne contiennent pas des panneaux routiers. Une fenêtre coulissante, ayant la même taille que le modèle, glisse sur l'image pour rechercher sa région la plus similaire. La fonction de similarité utilisée est l'erreur quadratique moyenne (MSE) est calculé comme est illustré dans l'équation suivante :

$$MSE = \frac{1}{MN} \sum_{y=1}^N [T(x, y) - I(x, y)]^2 \quad (2.29)$$

- $T(x, y)$  : L'intensité de l'image référence dans la position  $(x, y)$ ;
- $I(x, y)$  : L'intensité de l'image en entrée dans la position  $(x, y)$ ;
- $M$  et  $N$  : La largeur et la longueur de l'image respectivement.

La transformée de Hough a été utilisée par Garcia-Garrido et al. [32] afin de détecter les contours des panneaux, après ils sélectionnent que les contours fermés ce qui rend leur approche moins robuste aux bruits. Après ils utilisent un réseau de neurone pour la reconnaissance des panneaux. Leur approche est capable de détecter 97% des 435 panneaux de limitation de vitesse et 94% des 312 panneaux de danger dans un temps qui se varie de 20 ms/img jusqu'à 200 ms/img en fonction du nombre de contours traités. Même Miura et al [29] ont appliqué la transformée de Hough pour les panneaux circulaires.

Dans [33], les auteurs ont préféré d'utiliser la Transformation de Symétrie Radiale (TSR) pour détecter les panneaux de limitation de vitesse. Cette méthode est une variante de la transformée de circulaire de Hough est particulièrement destinée à détecter des occurrences possibles des panneaux circulaires. Les auteurs ont atteint un taux de détection de 96% pour 10% de faux positifs en 50 ms/img pour une image de taille  $320 \times 240 \text{ pixels}$ . Les auteurs de [34] ont également utilisé la TSR pour détecter d'autres formes géométriques comme octogone, carré et triangle. Cependant, ils ont atteint un taux de détection de 100% pour la forme en triangle, l'inconvénient majeur de cette méthode est qu'elle ne peut trouver que la taille et la position des formes, elle ne peut pas distinguer un panneau de céder le passage d'un panneau d'intersection.

Youssef, A. et al. [12] ont choisi la transformé de HOG (Histogram of Oriented Gradient) avec une fenêtre de détection  $40 \times 40$ . Ils ont adopté une architecture de trois tours de CNN avec le descripteur HOG. La base de données GTSRB est utilisé pour le premier apprentissage, DITS (The Data Set of Italian Traffic Signs) est utilisé pour le se-

cond tour d'apprentissage et les deux bases sont utilisées dans le troisième tour. Dans le but de réduire à la fois le nombre des faux positifs et le temps d'exécution, une étape de segmentation de couleur dans l'espace HSV améliorée est utilisée avant l'étape de détection. Les résultats établis par les auteurs sont illustrés dans le tableau 2.1.

TABLE 2.1 – Les résultats obtenus par Youssef, A. et al. [12]

Dataset	interdiction		Danger		obligation	
	Accuracy	Temps	Accuracy	Temps	Accuracy	Temps
GTSDDB	98.67%	231 ms	96.01 %	234 ms	90.43 %	243 ms
DITS	97.87 %	198 ms	98.12 %	197 ms	89.71 %	200 ms

Dans [76] les panneaux circulaires et polygonaux (non triangulaires) sont détectés par la transformation chinoise bilatérale TCB. La transformation chinoise fait toutefois l'hypothèse que l'objet est clair sur un fond sombre dans l'image, de sorte que les paires de points à considérer sont réduites aux couples (P1; P2) ayant des gradients convergents. Pour prendre en compte les cas de gradients d'orientations divergents, ils ont proposé plutôt de modifier le terme de phase de la TC. Deux alternatives sont testées : dans la première, les gradients divergents ont une contribution négative (c'est la TCB signée) et dans la seconde, ils ont la même contribution positive (c'est la TCB non signée). Comme les panneaux sont identifiables aussi par leurs couleurs ils ont combiné la sélection sur la forme et les couleurs selon deux approches :

- L'approche TC + Camshift, la TC sert à déterminer les centres de symétrie et d'initialiser la fenêtre du Camshift. La taille des objets ayant un indice de symétrie suffisant est déterminé en opérant une segmentation par l'algorithme du Camshift [80]. Elle est plus rapide mais elle rate plus de panneaux par rapport à la deuxième. En effet, sur les 81 panneaux rouges présents dans la base de test, 75% seulement sont détectés avec 22 faux positifs. le temps moyen est de 10 ms/image (contre 30 ms/image pour la TCB). Ces temps ne tiennent pas compte les prétraitements appliqués sur l'image, tels que le calcul du gradient et de son orientation, mais tiennent compte que la recherche de maxima locaux dans les accumulateurs.
- Une deuxième approche : un traitement par la TC ou TCB (signée ou non) sur les contours du canal rouge normalisé. Mais l'inconvénient majeur est la détection des panneaux de couleurs quelconques même la couleur verte. Sur les 92 panneaux présents dans la base de test, 86% sont détectés avec 25 faux positifs.

Le cas des panneaux triangulaires ne peut pas être traité par les transformées précédentes. Donc ils proposent une transformation spécifique pour les triangles qui combine la détection des sommets et du centre du triangle. En plus la complexité des algorithmes bivariés est en  $O(n^2)$  où  $n$  est le nombre de points de contour. Ce travail focalise sur la détection des panneaux et ne cherche pas à reconnaître leur sémantique.

Dans [17] un processus d'extraction des contours avec leurs tangentes associées de

précision subpixelique est effectué. Ils proposent trois algorithmes de détection d'ellipse, de triangle et de quadrilatère de type RANSAC, la forme finale sera retenue en fonction des degrés de compatibilité fournis par chacun des algorithmes. Ils utilisent la méthode présentée dans [81] et [82] pour estimer le centre de la forme en utilisant seulement trois points avec leurs tangentes. Plus de 80% parmi 1400 images de test sont correctement détectées, avec un faux positif vaut 5% dans un temps de traitement varie de 15 à 20 sec pour une image de taille  $1980 \times 1024 \text{ pixels}$ .

Qin, F. et al. [83] exploitent les descripteurs de Fourier (FD) pour décrire le contour, la raison principale de l'utilisation des FD est leur robustesse à la rotation, à l'échelle et à la translation, puis un processus d'appariement des FD est appliqué. Une base de données contenant plus de 20000 images a été créée en enregistrant des séquences sur plus de 350 km d'autoroutes suédoises et de routes urbaines. Cette base de données est utilisée pour évaluer l'approche proposée. Le taux de détection moyen atteint est de 77,08% sur 641 images au total.

Ruta et al. [2] ont proposé une variante de la DT (Distance Transform) appelée CDT (Colour Distance Transform), ils appliquent une comparaison entre l'image réelle et l'image modèle dans une représentation de la couleur discrète, pour faciliter la comparaison, pour chaque couleur discrète ils calculent une transformée en distance DT où les pixels ont une couleur discrète sont considérés comme des pixels caractéristiques et les autres pixels comme des pixels non caractéristiques. Le résultat de ce processus est illustré dans la figure 2.2.

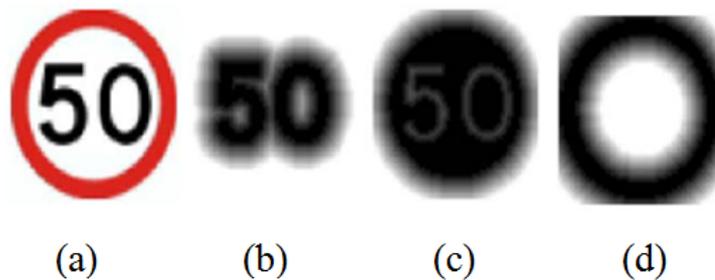


FIGURE 2.2 – CDT (colour distance transform) normalisée utilisée par [2] : (a) image originale dans la couleur discrète, (b) CDT noir, (c) CDT blanc, (d) CDT rouge. Les régions ombres signifient une petite distance.

Le problème de cette approche est l'absence d'une couleur donnée dans la CDT, pour pallier ce problème ils ont introduit une valeur maximale  $d_{max} = 10$  pixels, la CDT sera calculée pour chaque pixel  $p$  de l'image  $I$  comme suit :

$$d_{CDT}(I, p) \begin{cases} \frac{d_{CDT}(I, p)}{d_{max}} & \text{si } d_{CDT}(I, p) \leq d_{max} \\ 1 & \text{si } d_{CDT}(I, p) > d_{max} \end{cases} \quad (2.30)$$

Dans le but de reconnaissance de la forme du panneau, Qin et al. [3] utilisent le vecteur de la DtB proposé déjà par [84]. La DtB est la distance entre le contour extérieur et son rectangle englobant, alors pour chaque blob segmenté il y a quatre vecteurs DtB (left, right, top, bottom) comme est illustré dans la figure 2.3. Cette Distance est robuste aux translations et aux rotations, et pour la rendre robuste aux changements d'échelle, il faut zoomer le blob à  $36 \times 36$ . Ensuite, ce vecteur de DtB sera choisi comme une caractéristique pour classifier les blobs avec un SVM.

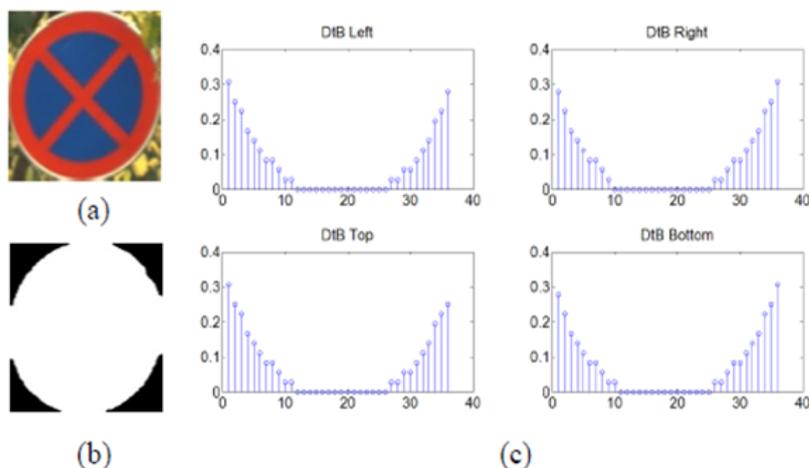


FIGURE 2.3 – DtB (Distance to border) proposé par [3], (a) le blob segmenté, (b) l'image binaire de (a), (c) DtB du blob

Ali et al. [4] ont proposé une autre approche pour reconnaître la forme du signe candidat; l'idée est de comparer le motif détecté avec le rectangle BoxOut qui est le plus petit rectangle qui englobe la forme. Un score d'intersections est calculé entre le contour du motif et les quatre lignes des BoxOuts montrées dans la figure 2.4.

Les scores sont calculés comme suit :

- score0 : Si il n y a aucune intersection;
- score1 : Si le contour occupe quelque points de la droite;
- score2 : Si le contour occupe presque un quart de la droite;
- score3 : Si le contour occupe presque la totalité de la droite.

Cette approche est capable de détecter 95,65% avec 2,17% de fausses alarmes. L'ensemble de données utilisé est constitué de 48 images de  $360 \times 270$  pixels contenant trois types de panneaux différents. L'inconvénient de cette approche est sa faiblesse aux occlusions et aux bruits.

Récemment, l'algorithme de détection de cercle EDCircles [85], est utilisé par [86] pour détecter les signaux de circulation. L'algorithme consiste à utiliser d'abord EDPF (Edge Drawing Parameter Free) pour détecter les contours dans les images en niveaux de gris, puis extraire les arcs des cercles et de combiner les arcs ayant un rayon similaire

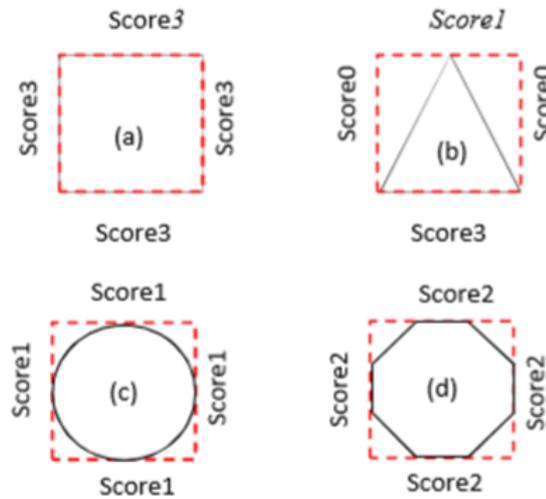


FIGURE 2.4 – Illustration de l'approche proposée par [4] (a) un rectangle, (b) un triangle, (c) un cercle, (d) un octogone.

ensemble, enfin, les cercles candidats sont validés. L'accuracy obtenue avec la base de données GTSDDB est de 93,78% et 0,99% de faux positifs pour les panneaux de danger, 75,51% d'accuracy avec 2,04% de faux positifs pour les panneaux d'obligation.

Table 2.2 résume les méthodes basées sur la forme. Les méthodes de détection précédentes n'utilisent pas la même base de données, ce qui nous empêche de les comparer. Les auteurs de [17] utilisent un grand ensemble de données avec de grandes images, ont atteint un taux de détection acceptable qui est encore très éloigné de l'application en temps réel. La méthode utilisée par [4] est en temps réel, mais la base de données utilisée est petite, ce qui nous empêche de la considérer comme une méthode très efficace.

TABLE 2.2 – Les méthodes basées sur la forme.

Auteurs	Méthode	Rappel(%)	Fausse alarmes	Base de données	Temps
[33]	Transformation de symétrie radiale	100	10	Des panneaux de limitation de vitesse de taille 320x240 pixels	50 ms/img
[4]	Score d'intersection avec le boxout	95.65	2.17%	48 images de taille 360x270 pixels	< 1 ms/img
[86]	EDCircles	interdiction : 93.78 obligation : 75.51	0.99% 2.04%	GTSDDB 36 ms/img	
[32]	Transformation de Hough	97 94	- -	435 image (limitation vitesse) 312 images (danger)	20 à 200 ms/img
[17]	3 algorithmes	80	5	1400 images de taille 1980x1024 pixels	15 à 20 sec/img
[24]	Distance de similarité	86	-	620 img avec 24 classes	-
[12]	HOG	Interdiction : 98.67 Danger : 96.01 Obligation : 90.43 Interdiction : 97.87 Danger : 98.12 Obligation : 89.71		GTSDDB  DITS	231ms 234 ms 243 ms 198 ms 197 ms 200 ms

### 2.2.1.3 Les méthodes avec apprentissage

Les méthodes calorimétriques et géométriques présentent une faiblesse commune à plusieurs facteurs tels que les changements d'éclairage, les occlusions, le changement d'échelle, la rotation et les translations. Cependant, ces problèmes pourraient également être traités en utilisant l'apprentissage automatique, mais cela nécessite une grande base de données annotées.

Les auteurs de [36] ont utilisé le détecteur Viola-Jones pour détecter les panneaux triangulaires. Le détecteur a été entraîné en utilisant environ 1000 images de qualité relativement médiocre. Le détecteur obtenu a atteint un rappel élevé (allant de 90% à 96%) en fonction de l'ensemble d'apprentissage et de la configuration du détecteur. Dans leurs expériences [36, 37], ils ont observé deux faiblesses principales du détecteur Viola-Jones :

- L'exigence d'un grand nombre d'images d'entraînement ;
- Un taux élevé de faux positifs.

Néanmoins, leurs recherches indiquent également que le détecteur Viola-Jones est robuste au bruit et il ne demande pas un ensemble d'apprentissage de haute qualité [87].

Bario et al. [88] ] ont proposé un classifieur en cascade où chaque entrée de la cascade est la région d'intérêt détectée par le classifieur précédent, l'algorithme Adaboost a été utilisé pour apprendre les classifieurs. Les chercheurs ont également proposé une stratégie de classification Forest-ECOC (Error Correcting Output Codes) pour surmonter le problème de multi-classification, l'idée étant d'intégrer plusieurs arbres dans le cadre ECOC. Les auteurs ont obtenu les résultats suivants, mais le temps de traitement n'est pas donné :

- les panneaux d'interdiction : 70% de rappel avec 3.65% de faux positifs.
- Les panneaux d'obligation : 60% de rappel avec 0.95% de faux positifs.
- Les panneaux de danger : 65% de rappel avec 2.25% de faux positifs.
- Panneaux de droit de passage : 75% de rappel avec 2.8% de faux positifs.

Priscariu et al. [39] ont utilisé le classifieur Adaboost basé sur le détecteur de Viola et Jones, suivi par un SVM avec le RGB normalisé. Le système est robuste au flou de mouvement grâce à la phase de suivi en 3D.

Chen, T. et al. [38] exploitent à la fois le Adaboost et le SVR (support vector regression) afin de détecter les panneaux. L'approche proposée est entraînée sur trois bases de données (GTSD, BTSD, STSD) en utilisant un processeur Intel Core-i7 4770 avec 8G de RAM. Cette approche n'est pas en temps réel, le temps de détection varie de 0,05 à 0,5 seconde et le temps d'apprentissage est de 16 minutes. Le rappel obtenu sur STSD est de 80,85% avec une précision de 94,52%, avec les bases GTSD et BTSD le rappel moyen obtenu est 99.93% et 97.03% respectivement.

De la Escalera et al. [1] ont utilisé des algorithmes génétiques dans l'étape de détection, ils ont appliqué une recherche parallèle dans différentes directions suivi d'un processus d'optimisation qui imite l'évolution naturelle et la sélection. Leur approche est robuste aux changements d'échelle, à la rotation, aux conditions météorologiques et à l'occlusion partielle, cependant elle n'est pas une application temps réel car l'algorithme prend 8.8 ms pour chaque génération et ils ont indiqué que le nombre maximal de générations est 51, sans calculer le temps de segmentation et classification. Fang et al. [40] ont utilisé les réseaux de neurones pour reconnaître la forme des panneaux, mais ce processus n'est pas linéaire avec un temps 2 sec/img.

L'histogramme des gradients orientés (HOG) proposé par Dalal et al. [89] pour la détection des piétons, est utilisé par Zaklouta et al [90] pour détecter les panneaux de danger en raison de sa robustesse aux changements d'échelle. Les descripteurs HOG sont utilisés comme une caractéristique pour entraîner le SVM, et pour améliorer sa précision, ils utilisent un opérateur morphologique (blackhat) pour filtrer les candidats détectés. La transformation de blackhat est définie comme la différence entre l'image de fermeture et l'image d'entrée, ils arrivent à réduire le nombre des faux positifs avec ce filtre.

Le descripteur HOG est également utilisé par Wang et al. [91] et ils utilisent LDA et SVM comme classifieur. L'approche proposée obtient des taux de rappel et de précision élevés avec la base de données GtSDB, cette approche est robuste aux mauvaises conditions d'éclairage, aux occlusions partielles et petites déformations. Le temps de traitement n'est pas indiqué et l'utilisation des ondelettes comme descripteur ralentit.

Dans la compétition "German Traffic Sign Detection Benchmark" présentée à IJCNN2013 (Conférence internationale conjointe sur les réseaux neuronaux), Hoben et al. [92] évaluent certaines approches de détection les plus populaires telles que le détecteur Viola-Jones basé sur les caractéristiques de Haar, un classifieur linéaire s'appuyant sur les descripteurs HOG et ils évaluent également un algorithme récent exploitant la forme et la couleur dans un schéma de vote Hough-like basé sur un modèle présenté dans [93]. Pour les trois algorithmes, ils ont obtenus les résultats suivants :

- Viola-jones : 98.8% pour les panneaux d'interdiction, 74.6% pour les panneaux de danger et 67.3% pour les panneaux d'obligation.
- HOG+LDA : 91.3% pour les panneaux d'interdiction, 90.7% pour les panneaux de danger et 69.2% pour les panneaux d'obligation.
- Hough-like : 55.3% pour les panneaux d'interdiction, 65.1% pour les panneaux de danger et 34.7% pour les panneaux d'obligation.

Salti, S. et al [94] proposent une approche basée sur l'extraction de régions d'intérêt plutôt que sur la détection avec fenêtres glissantes. Les auteurs testent leur approche sur une base de données de vérité-terrain qui contient 6580 images. L'ensemble de données est créé en utilisant 5 caméras pour localiser correctement les signes, cependant, utiliser 5 caméras n'est pas pratique pour une application en temps réel, car cela

augmente le temps d'exécution (au lieu de traiter une seule image, 5 images seront traitées). Ils ont atteint un accuracy de 78,21% pour les panneaux d'interdiction, 82,13% pour les panneaux de danger et 72,78% pour les panneaux obligatoires.

Le détecteur Integral Channel Features basé sur le descripteur HOG et les arbres de décision boostés est utilisé par [95]. Ils ont utilisé la base GTSDDB comme une base d'entraînement, ils ont obtenu avec la base de données GTSDDB un accuracy de 96,98% pour les panneaux d'obligation, 100% pour les panneaux de danger et 100% pour les panneaux d'interdiction. L'inconvénient de leur système est qu'il n'est pas scalable et ses performances se diminuent lorsqu'ils changent la base de test, avec la base BTSD ils ont obtenu un accuracy de 97,96% pour les signes d'obligation, 97,40% pour les panneaux de danger et 94,44% pour les panneaux d'interdiction. Ils n'ont pas donné le temps de test et d'entraînement de la phase de détection.

Pour résoudre le problème de la détection multiclassés des panneaux de signalisation, [79] ont présenté un framework capable de détecter rapidement les panneaux routiers, mais en haute résolution. Ils ont proposé trois nouvelles approches : premièrement, MN-LBP (Multi-block Normalized local binary pattern) et TMN-LBP (Titled MN-LBP) sont utilisés comme caractéristiques discriminantes pour exprimer efficacement les différentes classes des panneaux routiers. Deuxièmement, une structure arborescente appelée Split-Flow Cascade est conçue, ils utilisent les caractéristiques communes de plusieurs classes pour construire une structure pyramidale nommée SFC-tree. Troisièmement, pour concevoir un SFC-tree efficace, un AdaBoost de recherche commune (CF Ad-aBoost) est développé pour trouver des caractéristiques communes parmi les différents ensembles d'apprentissage. Ils ont atteint sur la base GTSDDB un accuracy de 100% pour les panneaux d'interdiction, 99,2% pour les panneaux de danger, 98,57% pour les panneaux d'obligation et 97,24% pour les autres signes avec un temps de traitement 192 ms.

Dans [96] un bon temps d'exécution a été atteint (6-8 fps). La solution présente une approche novatrice, appelée Categories-First-Assigned Tree (CFA-Tree) où ils intègrent la phase de détection et de classification dans un seul module. Ce nouveau système a une précision d'environ 93,5%, cependant, cet arbre de recherche ne peut détecter que trois catégories et a une faible efficacité avec les images de haute résolution [79].

En raison du succès de CNN dans la classification des panneaux de signalisation, les auteurs dans [15] ont proposé un ConvNet léger et optimisé avec une fenêtre coulissante pour détecter les panneaux de signalisation dans les images de haute résolution. Le taux de détection atteint avec la base GTSDDB est 99,89%. Le temps de traitement sur GPU (GeForce GTX 980) est de 26.506 ms (3772 images par seconde). Les résultats obtenus font de cette approche une application en temps réel. Les auteurs suggèrent que, en général, les ConvNets ne sont pas stables contre un bruit fort car ce sont des fonctions hautement non linéaires et que le vecteur de sortie peut changer de manière significative même avec de petites perturbations dans les entrées [15].

Wu et al [41] utilisent les réseaux neuronaux convolutionnels CNN pour localiser et

reconnaître les panneaux de signalisation, ils utilisent d'abord une machine vectorielle pour transformer l'image originale de RGB en échelle de gris pour éviter le problème de la différence de couleur due aux différentes conditions d'éclairage. Deuxièmement, ils utilisent la couche fixe dans le CNN pour localiser la région d'intérêt et les couches d'apprentissage sont utilisées pour extraire les caractéristiques discriminantes pour la classification. Avec la base GTSDb, ils ont obtenu 99.73% dans les panneaux de danger et 97.62% dans les panneaux d'obligation. Cependant, ils sont trop loin d'une application en temps réel car pour une architecture de 180-200 couches, pour chaque région d'intérêt le nombre de paramètres est 1290326 avec un temps d'exécution 0.128 secondes (128 ms), ils indiquent que le nombre de régions d'intérêt détectées dans une image est entre 5 et 20 régions. Alors le temps de détection pour une image de scène nécessite environ 2.5 secondes.

Des réseaux de neurones convolutionnels (CNN) en cascade sont utilisés par [42] pour réduire le nombre de faux positifs en utilisant le détecteur de LBP (Local Binary Pattern) associé au classifieur AdaBoost. Les résultats obtenus sur la base GTSDb sont 99.45% pour les panneaux d'interdiction, 96.50% pour les panneaux d'obligation, 98.33% pour les panneaux de danger. Le temps de traitement se varie entre 34 ms et 35 ms sur un processeur Intel Core 2 Duo 2.2 GHz.

La table 2.3 résume les méthodes de détection basée sur l'apprentissage. Les meilleurs rappels (> 99%) sont obtenus par [94], [95] et [91]. Cependant, ces méthodes peuvent-elles vraiment avoir un taux élevé et une précision élevée sur vérité terrain avec d'autres conditions complexes et en temps réel? De l'autre côté, [42] et [15] obtiennent de bons résultats avec une exécution en temps réel (35 ms). Habituellement, la plupart des méthodes de détection basées sur l'apprentissage atteignent un taux de détection supérieur à 95%. Il est donc temps de créer nouvelles bases de données plus compliquées, car les bases de données actuelles sont saturées.

TABLE 2.3 – Les méthodes de détection avec apprentissage.

Méthode	Base	Interdiction	Danger	Obligation	Tous les panneaux	Temps
[15]	GTSDDB	-	-	-	99.89%	26.506 ms
[88]	-	70% et 3,65% de FP	65% et 2,25% de FP	60% et 0,95% de FP	-	-
[36]	1000 images	-	de 90% à 96%	-	-	-
[38]	STSD GTSDDB BTSD	100% 97.78%	100% 99.88%	99.87% 93.45%	80.85%	0.05-0.5 sec
[92]	HOG+LDA Hough-like Viola-Jones	91.3% 55.3% 98.8%	90.7% 65.1% 74.6%	69.2% 34.7% 67.3%	- - -	- - -
[2]	GTSDDB	-	-	-	93.5%	Real time
[79]	GTSDDB	100%	92.2%	98.57%	-	192 ms
[95]	BTSD GTSDDB	94.44% 100%	97.40% 100%	97.96% 96.98%	- -	- -
[94]	verité terrain	78.21%	82.13%	72.78%	-	-
[91]	GTSDDB	99.88%	99.85%	99.85%	-	1.12-1.48 s
[90]	820 images	-	-	-	95%	247 ms
[42]	GTSDDB	99.45%	98.33%	96.50%	-	35 ms

### 2.2.2 Classification

Dans cette section, nous classifions quelques méthodes récentes et efficaces en deux parties : les méthodes d'apprentissage basées sur des caractéristiques extraits manuellement par un expert comme SIFT, SURF, LBP, HOG..., et les méthodes d'apprentissage profond (deep learning).

#### 2.2.2.1 les méthodes d'apprentissage basées sur des caractéristiques extraits manuellement

Fatin Zaklouta et al. [43] ont utilisé des descripteurs HOG (Histogram of Oriented Gradient) de différentes tailles et DT (Distance Transform) pour évaluer les performances des arbres Kd et Random Forests, les Random Forests sont plus robustes aux variations de background que les arbres Kd. L'accuracy de classification obtenue pour les Random Forest est de 97,2% avec les descripteurs HOG et de 81,8% avec Distance Transform, les arbres Kd atteignent respectivement de 92,9% et 67%.

Ellahyani, A. et al. [44] ont calculé le descripteur HOG dans l'espace de couleur HSI, puis les combinent avec les caractéristiques d'auto-similarité locale (LSS). Les auteurs préfèrent utiliser le Random Forest comme classificateur. L'accuracy dans la phase de classification est de 97,43% sur le GTSDb et 94,21% pour le système entier (la phase de détection et de classification) avec un temps de traitement 8-10 fps.

Les auteurs dans [97] ont comparé les performances de reconnaissance des panneaux routiers par l'être humain et les méthodes d'apprentissage de la machine, ils ont montré également les résultats d'un classificateur linéaire entraîné par LDA (Linear Discriminant Analysis), ils ont montré que les performances de LDA sont fortement liées aux caractéristiques choisies pour représenter l'image. Les auteurs ont obtenu de meilleurs résultats avec la représentation HOG2 avec une précision de 95,68%, 93,18% pour HOG1 et 92,34% pour HOG3.

L'analyse réalisée par Qu et al. [98] sur les bases de données d'apprentissage montre qu'il existe un déséquilibre dans la distribution des échantillons dans les classes des panneaux de signalisation. La plus grande classe peut contenir plus de 1000 images tandis que la classe la plus petite contient seulement plusieurs images. Pour pallier ce problème, les auteurs ont proposé une méthode de classification hiérarchique pour la reconnaissance des panneaux routiers. L'arbre de classification étant composé de deux couches, dans la première couche le classifieur Adaboost combiné avec ACF (Aggregate Channel Features) sont utilisés pour classer les panneaux en trois catégories selon leur forme géométrique. L'ACF est utilisé pour la représentation des caractéristiques où 10 canaux sont utilisés (trois canaux de couleur de l'espace colorimétrique RGB, les amplitudes de gradient, les cartes de gradient orientées thésix : horizontal, vertical, 30, 60, 120 et 150), puis ces caractéristiques sont utilisées pour trainer le classifieur Adaboos. Dans la deuxième couche le panneau est identifié avec le Random Forest entraîné sur

## Chapitre 2 : Etat de l'art

---

trois caractéristiques : HOG , LBP et HSV, ils ont atteint une précision de 95,97% sur GTSRB et de 97,94% sur STSD.

Yakimov [13] ont obtenu une reconnaissance en temps réel des panneaux de signalisation en utilisant la technologie de programmation multithread CUDA sur un GPU mobile Nvidia Tegra K1, qui contient 192 cœurs graphiques et 4 coeurs CPU de l'architecture ARM. Les auteurs ont proposé un nouvel algorithme GHT (Generalized Hough Transform) pour la classification des panneaux, ils comparent leurs résultats obtenus avec ceux de [92] et ils montrent que leur algorithme arrive à faire un bon compromis entre le temps d'exécution et la précision comme est illustré dans le tableau 2.4. La base de données allemande GTSRB a été utilisé mais seulement 9987 parmi 50000images ont été prises en compte.

TABLE 2.4 – Resultats obtenus par Yakimov [13]

Algorithme	Accuracy(%)	Temps(FPS)
Sliding window + SVM [92]	100	1
GHT avec preprocessing [13]	97.3	43
GHT sans preprocessing [13]	89.3	25
Viola-Jones[92]	90.81	15
HOG [92]	70.33	20

Les caractéristiques LBP et HOG sont utilisées par Li, C. et al [45] pour classifier le panneaux routiers. Le taux de précision atteint est de 95,16% pour HOG et de 95,38% pour LBP avec la base de données GTSRB. Dans la recherche de la caractéristique la plus discriminante qui peut représenter les panneaux, les auteurs dans [86] ont utilisé trois catégories de descripteur de fonctionnalité HOG, LBP et filtre Gabor comme caractéristique d'entrée de SVM, ils ont obtenu les résultats suivants :

- LBP : 93.36% dans 1.21 ms
- Gabor : 93.90% dans 2.31 ms
- HOG : 94.56 % dans 0.02 ms
- HOG+LBP : 95.24 % dans 1.24 ms
- HOG+Gabor :97.00% dans 2.40 ms
- Gabor+LBP : 95.17% dans 3.49 ms
- Gabor+LBP+HOG : 97.04% dans 3.60 ms

He, X., et Dai, B.[46] ont proposé une nouvelle variante de LBP (Local Binary Pattern) appelée MS-CSLBP (MultiScale Center Symmetry Local Binary Pattern) utilisée comme caractéristique locale et les coefficients de basse fréquence de DWT ( Discrete Wavelet Transform) en tant que caractéristique globale des panneaux de signalisation.

La grande différence entre LBP et CSLBP est que ce dernier remplace la valeur de pixel centrale par la valeur de pixel qui est symétrique au pixel central qui réduit la dimension des vecteurs caractéristiques à  $2^{N/2}$  plutôt que  $2^N$  dans LBP comme est illustré dans l'équation suivante :

$$CSLBP_R^N = \sum_{i=0}^{\frac{N}{2}-1} S(g_i - g_{i+p/2})2^i \quad (2.31)$$
$$S(x) = \begin{cases} 1 & x \geq \tau \\ 0 & x < \tau \end{cases}$$

Un seuil  $\tau$  est utilisé pour réduire l'influence du bruit. CSLBP peut réduire la dimension du vecteur de caractéristiques et son calcul est très simple par rapport à HOG et SIFT. Cependant, la caractéristique CSLBP de l'échelle unique n'est pas une caractéristique discriminante pour représenter les panneaux, pour résoudre le problème, les auteurs ont proposé de calculer CSLBP dans plusieurs échelles (rayon = 1,2 et 3). Le MS-CSLBP est ensuite utilisé comme caractéristique pour entraîner le SVM. Ils ont obtenu avec la base GTSRB un rappel de 97.67% dans un temps 121 ms.

SIFT est utilisé par Sathish, P. et al. [47], ils testent leur approche sur huit vidéos capturées en Inde. Le taux de rappel varie entre 75% et 100% avec un taux de faux positif qui ne dépasse pas 2%. Les auteurs dans [48] ont également utilisé SIFT combiné avec le sac de visuels pour construire le code book et ils utilisent le SVM pour la classification. La base de données utilisée pour l'évaluation est constituée de 130 images (50 \* 50 pixels), ils ont obtenu un accuracy de 93%. Pour exploiter la structure intrinsèque du codebook visuel, une nouvelle approche utilisant le codage parcellaire (group sparse coding) est proposée par [99]. Le taux de reconnaissance atteint par l'approche proposée est de 97,83% sur la base de données GTSRB.

Abdin et al. [49] ont utilisé le descripteur SIFT avec un réseau de neurones pour reconnaître les panneaux de signalisation. Ils ont créé une nouvelle base de données contenant 200 images capturées à partir des routes du Bangladesh dans différentes conditions météorologiques et d'éclairage. Le taux de vrai positif atteint est de 97% avec 3% de faux positifs. Ils ont de bons résultats, cependant l'exécution du temps n'est pas mentionné et leur base de données ne contient pas tous les types de panneaux de signalisation.

Han et al. [50] ont combiné SURF (Speeded Up Robust Features) avec la méthode de recherche K-NN (K-Nearest Neighbour) et ils ont proposé une nouvelle stratégie de sélection de caractéristiques. Les points d'intérêt non utiles sont éliminés en appliquant un seuil du déterminant de la matrice de Hesse, seuls les points d'intérêt ayant un déterminant plus important sont considérés. Afin de trouver de bons résultats, ils ont proposé une nouvelle stratégie de sélection des caractéristiques en calculant la pre-

mière et la deuxième distance minimale. Ensuite, la correspondance (matching) entre les points d'intérêt est calculé comme suit :

$$\frac{d_1}{d_2} < t_1 \quad \text{et} \quad d_1 < t_2 \quad (2.32)$$

$d_1, d_2$  : La première et la seconde distance minimale.

$t_1, t_2$  : Le seuil relatif et le seuil absolu.

Pour évaluer leur approche, ils ont créé une nouvelle base de données comprenant plus de 1200 images. Ils ont obtenu un taux d'erreur total de reconnaissance 12.81% et 0.99% du taux de fausse classification. L'approche proposée n'est pas robuste au flou, ils ont obtenu une fausse reconnaissance totale de 4% après élimination des images floues de l'ensemble de données.

Chen, L. et al [100] ont utilisé SURF comme une caractéristique pour classer les panneaux de signalisation, d'abord ils divisent les templates des panneaux en huit catégories selon la couleur, après ils utilisent le classifieur Adaboost pour réduire le temps du traitement. Ensuite, l'algorithme Approximate Nearest Neighbor (ANN) est utilisé pour l'étape de matching. L'accuracy obtenue est de 92,7% avec 200 images contenant 281 panneaux de signalisation.

Le descripteur SURF est également choisi par [4] en raison de son efficacité, sa rapidité et de sa robustesse aux changements d'éclairage par rapport à SIFT et PCA-SIFT. Les auteurs ont créé pour chaque classe des panneaux un template pour éliminer les points d'intérêt détectés en arrière-plan et conserver uniquement les points à l'intérieur du signe. Le taux de reconnaissance obtenu est de 97,72% avec 48 images.

Hoferlin, B et al [46] ont présenté une architecture pour la reconnaissance des panneaux circulaires, elle se compose de deux perceptrons multi-couche (MLP). La première couche utilise SIFT comme caractéristique d'entrée, la deuxième couche utilise SURF. La performance du système est évaluée sur une séquence de 30 minutes qui contient 133 panneaux de signalisation. Ils ont atteint un rappel de 96,4 %.

Une analyse comparative de trois techniques d'appariement de caractéristiques SIFT, SURF et BRISK (Binary Robust Invariant Scalable Key points) est présentée par [14]. Les auteurs ont créé une nouvelle base de données contenant 172 images classées en 32 catégories, après évaluation, ils ont observé que SIFT est mieux que SURF et BRISK. En temps d'exécution, BRISK est presque deux fois plus rapide que SIFT, mais ce n'est pas en temps réel. Les auteurs ont évalué le système selon deux différents scénarios : dans le premier scénario, les panneaux sont segmentés manuellement, alors que dans le second, les régions sont extraites automatiquement avec une segmentation. Les résultats obtenus par les auteurs démontrent que les performances de classification dépendent des résultats de l'étape de segmentation comme est illustré dans le

tableau 2.5 :

TABLE 2.5 – Résultats obtenus par Malik, Z. et al [14]

Méthode	Rappel(%)		temps (ms)
		segmentation manuelle	segmentation automatique
SIFT	100%	93.75%	9.03
SURF	93.75%	81.25%	6.02
BRISK	93.75	87.5%	4.3

Récemment, les auteurs dans [51] ont choisi aussi le descripteur SURF, le taux de rappel atteint est de 94,28% sur leurs base de données créé qui contiennent 179. Le temps de traitement obtenus se varie de 0.3-28 ms en fonction du nombre des points d'intérêt dans l'image qui est entre 20 et 85 points d'intérêt. Dans cette approche, les objets détectés dans une zone similaire à la scène sont fusionnés à un seul objet, de sorte que deux panneaux superposés peuvent être considérés comme un seul panneau, ce qui peut influencer sur les performances d'appariement.

Les méthodes d'apprentissage basées sur des caractéristiques extraits manuellement sont moins précises que les ConvNets, cependant, les deux méthodes ne sont pas évolutives et ne peuvent pas classer les panneaux d'un autre pays ou d'une nouvelle base de données. Pour pallier ce problème, les auteurs dans [101] utilisent les attributs visuels en raison de leur évolutivité, ils les combinent avec des réseaux bayésiens qui peuvent estimer la classe probable d'une nouvelle entrée en utilisant les observations. Le taux de classification obtenue pour chaque classe de GTSRB est de : 97,01% pour les panneaux de limitation de vitesse, 97,09% pour les panneaux d'obligation, 96,31% pour les panneaux de danger et d'autres classes.

Nous résumons les méthodes de classification avec les caractéristiques extraites manuellement dans le tableau 2.6. Cependant, toutes les méthodes permettent d'obtenir une bonne classification, il n'existe pas de méthode avec un taux classification élevé et moins de faux positifs en temps réel ce qui reste un champ de recherche ouvert.

TABLE 2.6: Les méthodes de classification des panneaux routiers avec les caractéristiques extraites manuellement.

Auteurs	Base	Méthode	Rappel(%)
Abedin et al.[49]	200 images	SURF+ANN	97
Aghdam et al.[101]	GTSRB	Attributs visuels avec réseaux Bayesian	98.04
Behloul et al. [4]	48 images	SURF	97.72
Berkaya et al. [86]	GTSRB	GABOR	93.90
		LBP	93.36
		HOG	94.56
		HOG+LBP	95.24
		HOG+GABOR	97.00
		Gabor+LBP	95.17
		HOG+LBP+GABOR	97.04
Chen et al. [100]	200 images	SURF+ANN	92.7
Ellahyani et al.[44]	GTSRB	HOG + random forest	97.43
Han et al.[50]	1200 images	SURF+KNN	96
He et al.[102]	GTSRB	MS-CSLBP+ DWT avec SVM	97.67
Hofer et al.[46]	vidéo	SIFT+SURF avec MLP	96.4
Houben et al.[92]	GTSRB	Sliding window + SVM	100
		Viola-Jones	90.81
		HOG	70.33
Hua et al.[48]	130 images	SIFT+bag of visual	93
Lasota et al. [51]	179 images	SURF	94.28
Li et al. [45]	GTSRB	HOG	95.16
		LBP	95.38
Liu et al. [99]	GTSRB	group sparse coding	97.83

Malik et al.[14]	172 images	SIFT	93.75
		SURF	81.25
		BRISK	87.5
Qu et al.[98]	GTSRB	ACF avec Adaboost	95.97
		STSD	97.94
Sathish et al.[47]	videos	SIFT	75 à 100
Stallkamp et al.[97]	GTSRB	LDA+HOG1	93.18
Tang et al.[103]	GTSRB	HOG	95.92
Yakimov et al.[6]	GTSRB	GHT avec preprocessing	97.3
		GHT sans preprocessing	89.3

---

### 2.2.2.2 Les méthodes d'apprentissage profond (deep learning)

Les caractéristiques extraites manuellement ont une puissance de représentation limitée et sont fortement liées à la connaissance d'expert, par conséquent, elles ne peuvent pas être discriminantes avec un très grand ensemble de données. Pour surmonter ce problème et pousser la performance de reconnaissance, des fonctionnalités profondes sont nécessaires. Pierre Sermanet et Yann Le Cun [55] utilisent le réseau convolutionnel (ConvNets) pour apprendre de manière non supervisée les caractéristiques invariantes du panneau de signalisation en utilisant des images en entrée de taille 32x32 pixels de la base de données GTSRB, ils atteignent un rappel de 98,97%, mieux que la performance humaine (98.81 % comme est indiqué dans [97]). Ils ont augmenté la capacité et la profondeur de leur réseau en ignorant les informations de couleur, ils ont établi un rappel de 99,17%. Les auteurs ont obtenu le meilleur résultat en ignorant la couleur, ils ont donc suspecté que les canaux de couleur normalisés soient plus informatifs que la couleur brute. les auteurs n'ont pas indiqué le temps de traitement, mais cette méthode est encore loin d'une application en temps réel vu le nombre élevé des paramètres utilisés (1437791 paramètres).

Les couches entièrement connectées dans le réseau neuronal convolutif CNN sont entraînées par rétropropagation, la sensibilité de rétropropagation et l'entraînement excessif de couches entièrement connectées peuvent rendre les performances de généralisation de CNN sous-optimales. Ordinairement, les auteurs utilisent CNN comme un extracteur de caractéristiques et un classifieur, c'est vrai, ils obtiennent des ré-

sultats impressionnants, cependant, ils utilisent un réseau vaste et complexe sur une énorme base de données. Les auteurs dans [104] proposent une nouvelle approche où le CNN fonctionne comme un extracteur de caractéristiques profondes, ce qui signifie que seules les huit premières couches sont conservées et éliminent les couches entièrement connectées. Extreme Learning Machine ELM est utilisé ensuite en tant que classifieur pour ses performances de généralisation. La méthode proposée prend 5-6 heures en apprentissage sans GPU et atteint un taux de reconnaissance de 99.40 % sans aucune augmentation et pré-traitement de données comme dans [54], mais cette méthode n'est pas robuste au flou de mouvement.

Qian, R. et al [56] utilisaient aussi CNN comme extracteur de caractéristiques et MLP (perception multi-couche) comme classifieur. En comparaison avec le Convnet classique, dans la couche max pooling de MPPS (max pooling positions) les auteurs n'utilisent pas les valeurs max, à l'inverse, ils utilisent leurs positions. Le MPPs consiste à coder chaque position de valeur maximale en binaire de 4 bits, puis à les concaténer pour obtenir la fonctionnalité MPP. La précision obtenue en utilisant MPPs est augmentée à 98,86% sur GTSRB.

Xie, K. et al [57] ont observé que 80% des signes mal classés ont la même couleur, la même forme et le même pictogramme. Pour surmonter ce problème, ils proposent un CNN en cascade en deux étapes. La première étape, CNN est formée sur l'étiquette de la classe, tandis que la deuxième étape est formée séparément sur les super classes selon la forme et le pictogramme. La précision de la méthode proposée est de 97,94% sur GTSRB et ils diminuent le nombre de faux positifs avec le CNN en cascade de 430 à 202. Le temps d'exécution n'est pas mentionné.

Une nouvelle architecture de ConvNet est proposée par [15], elle permet de réduire le nombre de paramètres de 27 %, 22 % et 3 % par rapport aux ConvNets utilisés par [105, 55, 16] respectivement. Ils proposent un compact ConvNet qui réduit le nombre de paramètres de 52% par rapport au premier ConvNet proposé. Pour améliorer la précision de la classification, ils ont également proposé une nouvelle méthode pour créer un Convnet optimal en sélectionnant un nombre optimal de ConvNets avec la plus grande précision possible et moins d'opérations arithmétiques de 88% et 73 % par rapport à [105, 16] respectivement. L'accuracy obtenue par leur méthode sur la base de données GTSRB est de 99.23% avec 2 ConvNets (ConvNet compact) et 99.61% avec seulement 5 ConvNets ce qui réduit considérablement le temps d'exécution comme illustré dans la table 2.7. De l'autre côté, [55] atteint un accuracy de 99.46% avec un ensemble de 25 ConvNets, [13] ont obtenu un accuracy de 99.65% en utilisant 20 ConvNets. Pour tester l'évolutivité et la performance de l'ensemble de données croisées du nouveau ConvNet proposé par [15], ils utilisent le ConvNet déjà entraîné sur GTSRB pour identifier les panneaux de la base de données BTSC, l'accuracy obtenue est 92.12%.

Eickeler et al.[52] ont utilisé une architecture simple de Deep Neural Network pour identifier les panneaux circulaires, ils ont atteint un taux de reconnaissance de 97,5% sur la base GTSRB. Deux autres architectures de CNN sont proposées par [12], l'archi-

TABLE 2.7 – Résultats obtenus par Aghdam, H. H. et al[15] comparé avec les résultats de [16]

Méthode	accuracy(%)	Temps d'exécution CPU (ms)	Temps d'exécution GPU (ms)
1 ConvNet[15]	-	12.96	1.06
1 compact ConvNet[15]	-	12.47	1.03
1 ConvNet[16]	-	14.47	1.45
5 ConvNet[15]	99.23	64.8	5.30
2 Compact ConvNet[15]	99.61	24.94	2.06
20 ConvNet[16]	99.65	289.4	29.0

teature échelle unique se compose de deux étages de couches convolutives et de deux couches locales entièrement connectées suivies d'un classifieur softmax. Dans l'architecture multi échelle, la sortie de la première couche convolutionnelle est considérée comme entrée à la fois pour la seconde couche convolutionnelle ainsi que pour la première couche entièrement connectée. Les deux architectures sont évaluées sur la base de données GTSRB et leur nouvelle base de données DITS (Data Set of Italian Traffic Signs). Le taux de rappel atteint sur GTSRB est de 97,2% pour l'architecture à échelle unique et de 98,2% pour le multi-échelle, et pour la base de données DITS le taux de précision atteint est de 93,1% pour une échelle unique et 95% pour une échelle multiple.

Dan Cireşan et al [53] utilisent une implémentation GPU d'un réseau neuronal convolutif (CNN), ils appliquent le pré-traitement sur les images d'entrée en redimensionnant toutes les images en 48x48 pixels et testent trois types de normalisation pour surmonter les variations de contraste élevées. Ils ont obtenu un rappel de 99,15% en utilisant un comité de perceptrons multicouches (MLP) entraîné avec les descripteurs HOG, le CNN est entraîné sur l'intensité des pixels bruts pour améliorer les performances de reconnaissance. Après avoir gagné la phase finale de la compétition de benchmark de reconnaissance des panneaux de signalisation allemand avec un taux de reconnaissance de 99.46 % dans [54], ils utilisent un MCDNN multi-colonnes Deep Neural Networks en faisant la moyenne des activations de sortie de plusieurs colonnes DNN.

Aghdam, H.H. et al [52] ont proposé une nouvelle architecture de CNN qui établit un nouveau record de classification 99.51%. L'architecture proposée réduit de 85% le nombre de paramètres et 88% de multiplications par rapport à [104]. Cette réduction

est obtenue en utilisant ReLU (Leaky Rectified Linear Units) [12] comme une fonction d'activation qui utilise une seule comparaison et une seule multiplication pour calculer la sortie.

Un nouveau record de classification des panneaux est à nouveau réalisé par Aghdam, H.H. et al [15], ils proposent une nouvelle variante de leur CNN précédent proposée dans [52]. Les auteurs remplacent les images couleur par des images au niveau de gris, ils suppriment la couche de transformation linéaire. Et pour augmenter la flexibilité, une couche entièrement connectée est ajoutée au réseau et ils réduisent la taille du premier et du deuxième noyau. De plus, les images d'entrée sont redimensionnées à 44x44 pixels pour réduire le temps de traitement. La nouvelle meilleure précision obtenue est de 99,55% avec CNN simple et 99,70% avec un ensemble de 3 CNN. Le nouveau CNN est temps réel avec un temps de traitement 0,7 ms par image.

Dans le tableau 2.8 nous résumons les méthodes de classification des panneaux routiers basées sur l'apprentissage profond.

TABLE 2.8 – Les méthodes de classification des panneaux routiers basées sur l'apprentissage profond.

Auteurs	Base	Méthode	Rappel (%)
Aghdam et al.[106]	GTSRB	Nouveau ConvNet	99.23
		Compact ConvNet	99.61%
	BTSC	ConvNet entraîné sur GTSRB	92.12%
Aghdam et al.[107]	GTSRB	CNN	99.51%
Aghdam et al.[15]	GTSRB	1 CNN	99.55%
		3 CNN	99.70%
Cirecsan et al.[54]	GTSRB	MCDNN	99.46%
Cirecsan et al.[53]	GTSRB	MLPs + HOG + CNNs	99.15%
Eickler et al.[52]	GTSRB	DNN	97.5%
Jin et al.[16]	GTSRB	20 ConvNet	99.65%
Qian et al.[56]	GTSRB	CNN+MLP	98.86%
Qu et al. [98]	GTSRB	ACF + Adaboost	95.97%
		STSD	97.94%
Sermanet et al.[55]	GTSRB	Convnets	98.97%
		performance humaine	98.8%
		Convnets sans couleur	99.17%
Xie et al.[57]	GTSRB	CNN en Cascade	97.94%
Youssef et al. [12]	GTSRB	CNN simple échelle	97.2%
Zeng et al. [104]	GTSRB	CNN+ELM	99.40%

### 2.3 Conclusion

Dans ce chapitre, nous avons présenté les méthodes récentes et efficaces de détection et de classification des panneaux de signalisation. Les méthodes de détection récentes atteignent un taux de détection variant de 90% à 100%, néanmoins, il est difficile de décider quelle méthode est la meilleure.

Pour obtenir un taux de classification élevé, il est nécessaire d'adopter des caractéristiques discriminantes et un classifieur puissant. Les performances des méthodes de classification ont été renforcées par le deep learning ce qui permet d'atteindre un taux de classification plus de 99%. Les bases de données disponibles sont saturé car les chercheurs arrivent à atteindre des taux >99% sur les bases de données actuelles mais pas sur vérité terrain, donc une nouvelle base de données universel et plus complexe est indispensable .

Les méthodes de détection et de classification des panneaux routiers ont atteint un taux de précision élevé, cependant, elles sont encore loin d'une application ADAS temps réel où le panneau devrait être détecté et classé en temps réel.

La question qui se pose est la suivante : les méthodes récentes de détection et de classification des panneaux peuvent-elles prouver les mêmes performances dans les applications du monde réel ou avec d'autres bases de données de vérité terrain? Peuvent-elles prouver avec les smart phones le même temps obtenu sous l'environnement de CPU et de GPU? Enfin, un système universel de reconnaissance des panneaux de signalisation reste un domaine de recherche ouvert.

## **Chapitre 3**

### **Localisation des panneaux routiers**

### 3.1 Introduction

Le but de la localisation des panneaux de signalisation routière est de déterminer les régions d'intérêt où se trouve un candidat potentiel de signalisation routière. Le problème de localisation des panneaux de signalisation peut également être considéré comme un problème de segmentation d'image lorsqu'il existe deux classes : le panneau de signalisation et le background. Le problème de segmentation reste ouvert, surtout dans le cadre d'analyse d'images réelles riches en détails et en contenu, une grande variété de régions de nature uniforme ou texturée comme dans le cas des images contenant les panneaux routiers. En raison de variation d'éclairage et des conditions météorologiques, la localisation des panneaux de signalisation en utilisant des informations de couleur, en particulier dans les images en plein air est une tâche très difficile. Néanmoins, la segmentation correcte de ces images aide à reconnaître le contenu et la forme du panneau dans les étapes ultérieures.

Dans ce chapitre, nous présentons les différentes méthodes de segmentation, les espaces de couleurs et les modèles visuelles ainsi que notre contribution dans la segmentation pour la localisation des panneaux routiers.

### 3.2 Background

#### 3.2.1 Les méthodes de segmentation

La segmentation d'image est le partitionnement d'une image en régions ou catégories, qui correspondent à différents objets ou parties d'objets dans l'image. Chaque pixel dans l'image est affecté à l'une de ces catégories. Une bonne segmentation est généralement celle dans laquelle :

- les pixels de la même catégorie ont des niveaux de gris à valeurs multivariées similaires et forment une région connectée,
- les pixels voisins qui sont dans des catégories différentes ont des valeurs différentes.

La segmentation est souvent l'étape critique dans l'analyse d'image, si la segmentation est bien faite, toutes les autres étapes de l'analyse d'image sont simplifiées, ainsi les performances de l'étape de reconnaissance de forme est fortement liés à la qualité de la segmentation. Mais, comme nous le verrons, le succès n'est souvent que partiel lorsque des algorithmes de segmentation avec seuillage (thresholding) sont utilisés.

Une façon de définir la segmentation de l'image est la suivante [108]. Formellement, un ensemble de régions  $R_1, R_2, \dots, R_n$  est une segmentation de l'image  $R$  en  $n$

régions si :

$$U_{i=1}^n R_i = R \quad (3.1)$$

$$R_i \cap R_k = \emptyset, i \neq k \quad (3.2)$$

$$R_i \text{ est connecté, } i = 1, 2, 3, \dots, n \quad (3.3)$$

Il y a un prédicat P qui mesure l'homogénéité de la région :

$$P(R_i) = \text{VRAI}, i = 1, 2, 3, \dots, n \quad (3.4)$$

$$P(R_i \cup R_k) = \text{FAUX}, i \neq k \text{ et } R_i \text{ adjacent à } R_k \quad (3.5)$$

Les conditions ci-dessus peuvent être résumées comme suit : la première condition implique que chaque point d'image doit être dans une région. Cela signifie que la segmentation ne doit pas se terminer avant que chaque point soit traité. La deuxième condition implique que les régions ne se chevauchent pas, tandis que la troisième condition détermine que les régions sont composées de pixels contigus. Enfin, la quatrième condition détermine quel type de propriétés les régions segmentées doivent avoir, par exemple, des niveaux de gris uniformes et exprime la maximalité de chaque région dans la segmentation [109].

Selon la classification proposée par Fu et Mui [110], les techniques de segmentation sont classées en trois catégories : (1) Thresholding ou clustering, (2) les méthodes basées sur la région et (3) les méthodes basées sur les contours.

### 3.2.1.1 Thresholding ou clustering

- Thresholding : Dans le seuillage, les pixels sont affectés à des catégories en fonction de la plage de valeurs dans laquelle un pixel se trouve. Le seuillage est la méthode de segmentation la plus simple et la plus couramment utilisée. Étant donné un seul seuil,  $t$ , le pixel situé à la position  $(i, j)$ , avec la valeur de l'intensité  $f_{i,j}$ , est affecté à la catégorie 1 si :  $f_{i,j} \leq t$ , sinon elle est affecté à la catégorie 2.

Dans des conditions contrôlées, si la réflectance de surface des objets ou régions à segmenter est uniforme et distincte du fond et que la scène est uniformément éclairée, alors l'image résultante contiendra des régions homogènes avec des limites bien définies qui conduisent généralement à un bimodal ou à plusieurs histogrammes modaux. Néanmoins, dans de nombreux cas, le niveau de fond n'est pas constant et le contraste des objets se varie au sein de l'image. Dans de tels cas, un seuil qui fonctionne bien dans une zone de l'image peut mal fonctionner dans d'autres zones. Ainsi, il est commode d'utiliser un seuil qui varie lentement en fonction de la position dans l'image [109]. Un seuil dynamique

a été proposé par Chow an Kaneko [111], qui découpe l'image en sous-images rectangulaires et calcule le seuil pour chaque sous-image. Cependant, une sous-image peut ne pas avoir de seuil si son histogramme de niveau de gris n'est pas bi-modal, et alors ces sous-images reçoivent des seuils interpolés provenant de sous-images voisines. Enfin, l'image entière est seuillée en utilisant les seuils séparés pour chaque sous-image.

Le succès de cette approche dépend de l'existence de seuils appropriés et de la possibilité de déduire les seuils de l'histogramme de l'image. Cependant, ceci n'est possible qu'avec un ensemble restreint d'images dans l'hypothèse d'un environnement contrôlé. En tant que méthodes pour aborder les images naturelles comme les images des panneaux routiers, où la variation de l'éclairage, le bruit et la texture sont présents, ils deviennent inadéquats.

- Clustering : le regroupement est un processus par lequel un ensemble de données est remplacé par des clusters, qui sont des collections de points de données qui ont les mêmes caractéristiques. Il est clair de penser à la segmentation de l'image comme un regroupement, regroupant les pixels qui ont la même couleur et / ou la même texture. Les méthodes de classification peuvent être divisées en deux types de base : le regroupement hiérarchique et le regroupement partiel.

Le regroupement hiérarchique procède successivement en fusionnant des groupes plus petits en groupes plus grands (algorithmes agglomératifs), ou en divisant des groupes plus grands (algorithmes diviseurs). Les méthodes de regroupement diffèrent dans la règle selon laquelle il est décidé quels deux petits groupes sont fusionnés ou quel groupe est divisé. Le résultat final de l'algorithme est un arbre de clusters appelé dendogramme, qui montre comment les clusters sont liés. En coupant le dendogramme à un niveau désiré, on obtient un regroupement des éléments de données en groupes disjoints.

D'un autre côté, le clustering partitionnel tente de décomposer directement l'ensemble de données en un ensemble de clusters disjoints. Une fonction objective exprime à quel point une représentation est bonne, puis l'algorithme de clustering essaye de minimiser cette fonction afin d'obtenir la meilleure représentation. Généralement, les critères globaux impliquent de minimiser une mesure de dissimilarité pour les échantillons dans chaque cluster (minimise l'intraclasse), tout en maximisant la dissimilarité entre les différents groupes (maximiser l'interclasse). La méthode de mise en clusters partielles la plus couramment utilisée est l'algorithme K-means [112] [113], dans lequel la fonction de critère est la distance au carré des éléments de données de leurs centroïdes les plus proches.

Les méthodes de clustering, même en tant que méthodes de seuillage, sont globales et ne conservent pas les informations de position. L'inconvénient majeur de ceci est qu'il est invariant au réarrangement spatial des pixels, ce qui est un aspect important de ce que l'on entend par segmentation. Les segments résultants ne sont pas connectés et peuvent être largement dispersés. Certaines tentatives ont été faites pour introduire de telles informations en utilisant des coordonnées de pixels en tant que caractéristiques. Cependant, cette approche tend à entraîner la fragmentation de vastes régions et les résultats ne sont pas meilleurs que

ceux qui n'utilisent pas l'information spatiale. La nécessité d'incorporer une certaine forme d'information spatiale dans le processus de segmentation a conduit au développement de méthodes dans lesquelles les pixels sont classés en fonction de leur contexte ou de leur voisinage [109].

### 3.2.1.2 Les méthodes basées sur la région

L'approche région tente d'isoler des zones d'images homogènes selon un ensemble de caractéristiques donné. Nous introduisons dans cette section deux méthodes classiques basées sur les régions : la croissance des régions et split-and-merge.

- La croissance des régions est : est l'un des algorithmes de segmentation par région les plus simples et populaires. Il commence par choisir un point de départ ou un pixel de départ. La manière la plus habituelle est de sélectionner ces graines en choisissant aléatoirement un ensemble de pixels dans l'image, ou en suivant a priori la direction de balayage de l'image. Ensuite, la région se développe en ajoutant successivement des pixels voisins similaires, selon un certain critère d'homogénéité, augmentant pas à pas la taille de la région. Ce critère peut être, par exemple, d'exiger que la variance d'une caractéristique à l'intérieur de la région ne dépasse pas un seuil, ou que la différence entre le pixel et la moyenne de la région soit faible. Le processus de croissance est poursuivi jusqu'à ce qu'un pixel non suffisamment similaire pour être agrégé soit trouvé. Cela signifie que le pixel appartient à un autre objet et que la croissance dans cette direction est terminée. Lorsqu'il n'y a pas de pixel voisin similaire à la région, la segmentation de la région est terminée. Le suivi de cette procédure donne l'impression que les régions à l'intérieur des objets grandissent jusqu'à ce que leurs limites correspondent aux contours de l'objet.
- Split and merge : l'une des propriétés fondamentales de la segmentation est l'existence d'un prédicat  $P$  qui mesure l'homogénéité de la région. Si ce prédicat n'est pas satisfait pour une région, cela signifie que cette région est inhomogène et devrait être divisée en sous-régions. D'un autre côté, si le prédicat est satisfait pour l'union de deux régions adjacentes, alors ces régions sont collectivement homogènes et devraient être fusionnées en une seule région. Une façon de travailler vers la satisfaction de ces critères d'homogénéité est l'algorithme split-and-merge [114]. Cette technique consiste, comme leur nom l'indique, de deux étapes de base. Tout d'abord, l'image est récursivement divisée jusqu'à ce que toutes les régions vérifient un critère d'homogénéité. Ensuite, dans une deuxième étape, toutes les régions adjacentes sont réassemblées de manière à ce que les régions résultantes satisfassent au critère d'homogénéité [109].

### 3.2.1.3 Les méthodes basées sur les contours

La dernière classe de procédés pour la segmentation d'image est liée à la détection des transitions de luminance entre les régions, c'est-à-dire les limites (lignes ou arêtes). L'importance fondamentale des informations de ligne et de contour dans les systèmes biologiques et de vision par ordinateur est reconnue depuis longtemps. En effet, les preuves biologiques montrant la détection des contours jouant un rôle central dans les premiers stades de la perception visuelle chez les mammifères (vision de bas niveau), comme le système visuel humain, ont souvent motivé son adoption par les chercheurs en traitement d'images. Les caractéristiques locales, telles que les lignes et les arêtes, peuvent décrire la structure d'une scène de manière relativement indépendante de l'éclairage.

Les techniques de segmentation de l'image basées sur la détection des contours sont utilisées depuis longtemps, depuis les premiers travaux de Roberts et al [115]. Bien que diverses méthodes de détection des contours aient été suggérées, il existe deux approches locales de base : la différenciation de premier ordre et de second ordre. Le fléau de toutes ces méthodes, cependant, est le bruit. Les contours, par définition, varient rapidement dans l'espace et ont donc des composantes significatives à des fréquences spatiales élevées. C'est aussi, malheureusement, la caractéristique du bruit, et donc tout opérateur de gradient qui répond bien à la présence d'une arête répondra également bien à la présence de bruit ou de textures signalant ainsi de faux arêtes.

- Premier ordre : un masque de gradient (Roberts [115] et Sobel [116] sont des exemples bien connus) est convolué avec l'image pour obtenir le vecteur gradient  $\nabla f$  associé à chaque pixel. Les contours sont les endroits où la magnitude du vecteur de gradient  $\|\nabla f\|$  est un maximum local suivant la direction du vecteur gradient  $\varnothing(\nabla f)$ . A cet effet, la valeur locale de l'amplitude du gradient doit être comparée aux valeurs du gradient estimées le long de cette orientation et à une distance unitaire de chaque côté du pixel. Après ce processus de suppression non-maximale, les valeurs des vecteurs de gradient restants sont seuillées, et seuls les pixels ayant une amplitude de gradient supérieure à ce seuil sont considérés comme des pixels de contour.

L'opérateur Sobel [116] a introduit une pondération des moyennes locales des deux côtés du pixel central. Plusieurs travaux ont cherché l'optimisation de ce facteur de pondération. Canny [117] a proposé le filtre dérivé de Gauss comme un filtre quasi-optimal par rapport à trois critères de recherche de contour : (a) bonne localisation du contour, (b) une seule réponse à un seul contour et (c) probabilité élevée de détection de vrais points de contour et faible probabilité de détection des faux points non-bord. Deriche [118], sur la base des critères de Canny, a implémenté un filtre avec une réponse impulsionnelle similaire à celle de la dérivée de Gauss, mais qui se prête à une implémentation directe comme filtre récursif.

### 3.2.2 Les espaces de couleur

Un espace colorimétrique est un modèle mathématique représentant l'ensemble des couleurs, chaque couleur est associée à des coordonnées déterminant un point précis dans cet espace de couleur. Différents espaces de couleur ont été proposés dans la littérature, chacun avec ses propres propriétés, avantages, limites et domaines d'application. Nous présentons dans la suite les différents espaces de couleur utilisés dans le domaine de la reconnaissance des panneaux routiers.

#### 3.2.2.1 Espace RGB

L'espace des couleurs primaires RGB (Red Green Blue), également appelé RVB en français (Rouge Vert Bleu), est calqué sur notre perception visuelle. Il utilise trois couleurs de base : le rouge ( $\gamma=700$  nm), le vert ( $\gamma=546$ nm) et le bleu ( $\gamma=435,8$ nm). Le modèle RGB peut être représenté comme un cube de trois dimensions comme est montré dans la figure 3.1, avec le rouge, le vert et le bleu sont les coins sur leurs axes. Le noir est l'origine et le blanc est au coin opposé. Le niveau de gris est sur la ligne entre le coin du noir et le coin de blanc [119, 120].

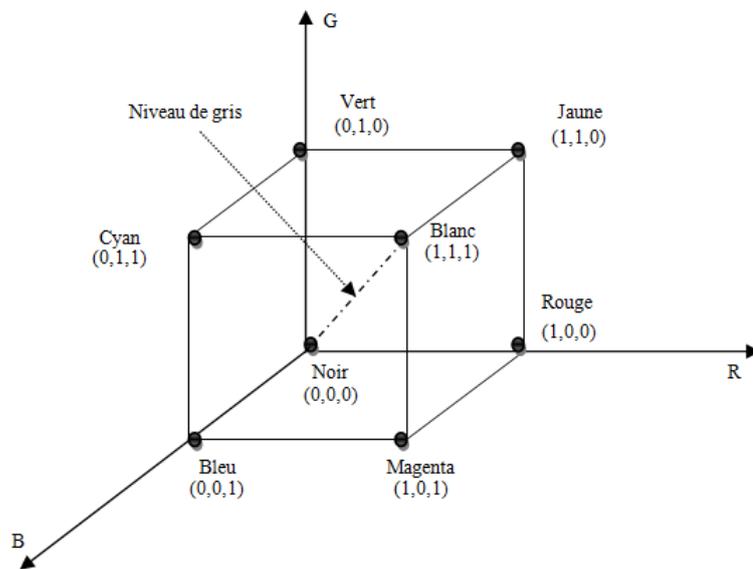


FIGURE 3.1 – Le Cube représentant le modèle RGB

La raison d'utiliser l'espace RGB est d'économiser le temps de calcul des conversions entre espaces de couleur. Pour éviter les effets des variations de luminosité suivant l'heure, la saison ou les conditions météorologiques, il faut fixer les seuils  $\alpha$ ,  $\beta$ , et  $\gamma$ , le plus souvent ces seuils sont fixés après des expérimentations sur un échantillon d'images.

### 3.2.2.2 Espace HSI

Dans l'espace colorimétrique HSI, la saturation, la teinte et l'intensité sont utilisées comme axes de coordonnées. La figure 3.2 montre une représentation possible de l'espace colorimétrique HSI. Cet espace de couleur est bien adapté pour le traitement des images en couleur et pour la définition visuelle des caractéristiques locales interprétables. Une couleur  $q = (R, G, B)$  est donnée dans l'espace colorimétrique RGB. La teinte  $H$  de la couleur  $q$  caractérise la couleur dominante contenue dans  $q$ . Le rouge est spécifié en tant que "couleur de référence". A cause de cela,  $H = 0^\circ$  et  $H = 360^\circ$  correspondent à la couleur rouge, formellement, ils sont donnés par :

$$H = \begin{cases} \delta & B \leq G \\ 360^\circ - \delta & B > G \end{cases} \quad (3.6)$$

Avec

$$\delta = \arccos\left(\frac{(R - G) + (R - B)}{2\sqrt{(R - G)^2 + (R - B) * (G - B)}}\right) \quad (3.7)$$

La saturation  $S$  de la couleur  $q$  est une mesure de la pureté de la couleur. Ce paramètre dépend du nombre de longueurs d'onde qui contribuent à la perception de la couleur. Plus la gamme des longueurs d'onde est large, plus la pureté de la couleur est faible. Plus la plage des longueurs d'onde est étroite, plus la pureté de la couleur est élevée. Le cas extrême  $S = 1$  est vrai pour une couleur pure et le cas extrême  $S = 0$  pour une couleur achromatique [121].  $S$  est calculé comme suit :

$$S = 1 - 3 \cdot \frac{\min(R, G, B)}{R + G + B} \quad (3.8)$$

L'intensité  $I$  de la couleur  $q$  correspond à la luminosité relative (dans le sens d'une image de niveau gris). Le cas extrême  $I = 0$  correspond à la couleur noire. L'intensité est définie conformément à

$$I = \frac{R + B + G}{3} \quad (3.9)$$

L'un des avantages de l'espace couleur HSI est la séparation des couleurs et l'in-

formation achromatique. L'existence de singularités est un inconvénient pour l'espace HSI. De plus, il faut noter que le contenu de l'information et la fiabilité du calcul de la teinte et de la saturation dépend de la luminosité. Dans les couleurs achromatiques, ni la teinte ni la saturation est définie [121].

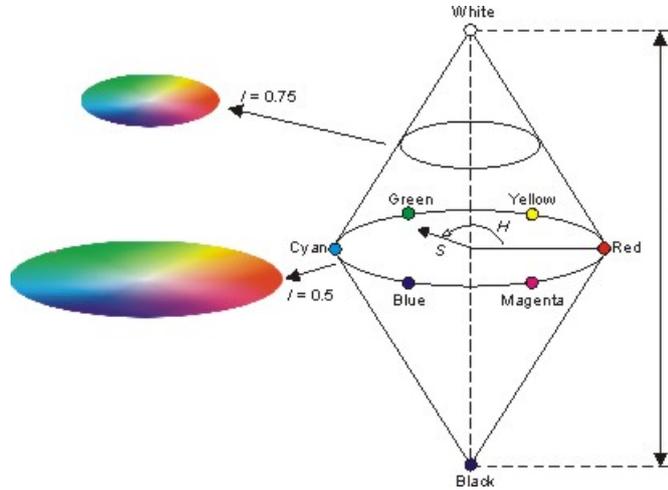


FIGURE 3.2 – L'espace de couleur HSI.

### 3.2.2.3 Espace HSV

Comme dans l'espace colorimétrique HSI, la teinte, la saturation et la valeur de luminosité sont utilisés comme axes de coordonnées. En projetant le cube de RGB le long des diagonales du blanc au noir, on obtient un hexaèdre qui forme la partie supérieure de la pyramide HSV. La dénomination de cette représentation vient de la traduction anglaise de teinte, saturation, valeur (luminance) : Hue, Saturation, Value [120].

- La teinte : est le nom de la couleur, c'est la longueur d'onde dominante. On peut calculé la teinte à partir de l'espace RGB en utilisant les équations suivantes :

$$H = \begin{cases} \frac{60 * (G - B)}{V - \text{Min}(R, G, B)} & G \geq B \\ 360 + \frac{60 * (G - B)}{V - \text{Min}(R, G, B)} & G < B \end{cases} \quad (3.10)$$

$$H = 120 + \frac{60 * (B - R)}{V - \text{Min}(R, G, B)} \quad G = V \quad (3.11)$$

$$H = 240 + \frac{60 * (R - B)}{V - \text{Min}(R, G, B)} \quad B = V \quad (3.12)$$

- La saturation : en indice de pureté représente l'inverse du degré de dilution de la

couleur dans la lumière blanche. La saturation est calculée comme suit :

$$S = \begin{cases} \frac{V - \text{Min}(R, G, B)}{V} & V \neq 0 \\ 0 & V = 0 \end{cases} \quad (3.13)$$

- La luminance : est l'intensité de la lumière achromatique. Elle est mesurable et additive. L'unité de brillance est le candela par mètres carrés ( $cd.m^{-2}$ ) dont l'unité correspond à 10 nits :  $1 cd^{-2} = 10 nits$ . la valeur de la luminance est donnée par :  $V = \text{Max}(R, G, B)$

Ce modèle est appelé également « modèle du peintre » en référence aux méthodes de peintures : le peintre commence par choisir une couleur (H) puis il ajoute du blanc pour désaturer la couleur (S) et ajoute du noir pour dévaluer la couleur (V) [120].

Le modèle HSV est un modèle avec les coordonnées cylindriques. La figure 3.3 illustre une représentation pour ce modèle. La teinte (H) est représentée par l'angle  $\theta$ , qui varie de  $0^\circ$  à  $360^\circ$ . La saturation est représentée par le rayon S, elle varie de 0 à 1 et plus la saturation est faible plus l'image sera grisée et apparaîtra fade. L'intensité sera représentée sur l'axe I, elle varie de 0 (noir) jusqu'à 1 (blanc) [119, 120].

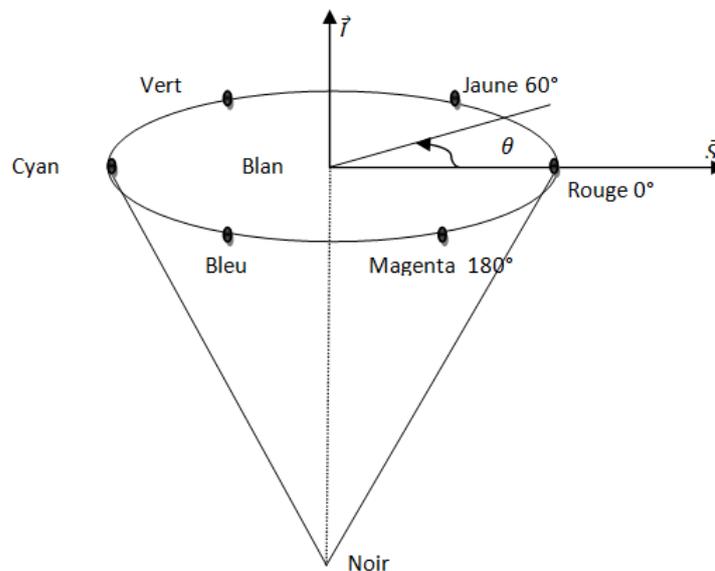


FIGURE 3.3 – Pyramide HSV : représentation de la couleur en fonction de la teinte, de la saturation et de la luminance.

#### 3.2.2.4 Espace CIE lab

La CIE (Commission Internationale de l'Éclairage) est un espace colorimétrique uniforme [121] utilisé principalement pour décrire les couleurs corporelles des composants (matériaux non luminance). Le **lab** représente une composante de luminance

**l**, et deux composantes de chrominance **a** et **b**.

La composante *a* représente deux couleurs; vert et rouge respectivement aux extrémités négatives ( $-a$ ) et positive ( $+a$ ). La composante *b* représente également deux couleurs, bleu et jaune respectivement aux extrémités négatives ( $-b$ ) et positive ( $+b$ ) [121]. Ce qui précède est illustré à la figure 3.4.

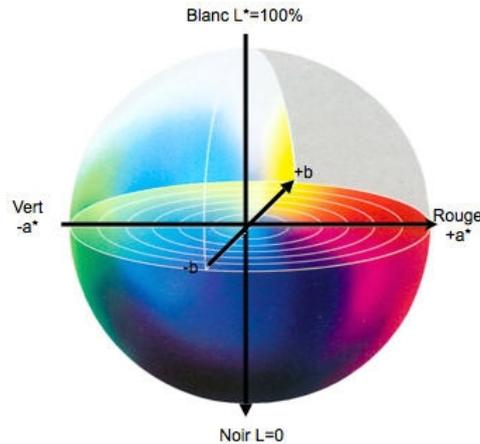


FIGURE 3.4 – L'espace de couleur CIE lab

La conversion de l'espace colorimétrique RGB à l'espace CIE lab est un processus double. Premièrement, l'espace colorimétrique RGB est converti en un espace colorimétrique intermédiaire XYZ. Ensuite, les composants intermédiaires sont utilisées pour la conversion vers l'espace colorimétrique CIElab. Les équations (3.14)-(3.16) représentent les formules mathématiques pertinentes utilisées.

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = (M) \begin{pmatrix} R \\ G \\ B \end{pmatrix} \quad (3.14)$$

Où

$$(M) = \begin{bmatrix} S_r X_r & S_g X_g & S_b X_b \\ S_r Y_r & S_g Y_g & S_b Y_b \\ S_r Z_r & S_g Z_g & S_b Z_b \end{bmatrix} \quad (3.15)$$

$$\begin{cases} l = 116 * Y - 16 \\ a = 500 * (X - Y) \\ b = 200 * (Y - Z) \end{cases} \quad (3.16)$$

### 3.2.2.5 Espace YUV

Les systèmes de télévision PAL et SECAM, développés en Allemagne et en France, utilisent l'espace de couleur YUV pour la transmission. Dans l'espace de couleur YUV, U correspond à la différence de couleur rouge-bleu et V à la différence de couleur vert-magenta. Y correspond aux moyennes pondérées de rouge, vert et bleu [121]. La transformation de l'espace RGB en YUV est calculé comme suit :

$$\begin{pmatrix} Y \\ U \\ V \end{pmatrix} = \begin{pmatrix} 0.299 & 0.587 & 0.114 \\ -0.418 & -0.289 & 0.437 \\ 0.615 & -0.515 & -0.100 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix} \quad (3.17)$$

### 3.2.3 Les modèles d'attention visuelle

Chaque seconde, un flux riche de données visuelles entrent dans l'œil, pour laquelle le traitement immédiat, sans aucun mécanisme astucieux pour réduire la quantité d'informations est extrêmement ardue. Le mécanisme d'attention visuelle (VAM) ou bien la saillance visuelle est un concept de psychologie cognitif qui fait ressortir certains stimuli d'une scène par rapport à leurs voisins et ignore automatiquement les informations superflues et attire notre attention aux objets les plus significatifs.

Les régions saillantes d'une scène attirent immédiatement notre attention. Détecter ces régions dans le champ de vision a une complexité de calcul qui rend difficile même pour les cerveaux, sans parler d'un ordinateur. La sélection des régions saillantes est possible grâce aux mécanismes d'attention visuelle. Plusieurs travaux ont été réalisés, frintrop a proposé un modèle appelé détection d'objets visuels avec le système d'attention computationnelle visuelle (VOCUS) [122] qui était une implémentation réussie de la vision du robot. Au cours de la même période, Walther et al. [123] s'appuyaient uniquement sur l'attention basée sur la saillance ascendante pour accomplir la reconnaissance d'objet. D'autres applications sur la détection et la reconnaissance d'objets ont été proposées comme [124], [125].

Selon Borji et al [126] les modèles d'attention visuelle sont classés en fonction de leur mécanisme pour obtenir la saillance.

- Les modèles cognitifs : sont des modèles biologiquement inspirés comme le fameux modèle proposé par Itti et al [5]. Il utilise trois canaux de caractéristiques couleurs, intensité et orientation. Ce modèle a été la base de modèles ultérieurs et le standard de référence pour la comparaison. Le Meur et al. [127] ont proposé une approche basée sur la structure du système visuel humain. Les fonctions de sensibilité aux contrastes, la décomposition perceptuelle, le masquage visuel et les interactions centre-environnement sont quelques-unes des fonctionnali-

tés implémentées dans ce modèle. Plus tard, Le Meur et al. [128] ont étendu ce modèle au domaine spatio-temporel en fusionnant des informations achromatiques, chromatiques et temporelles. Dans ce nouveau modèle, les caractéristiques visuelles précoces sont extraites de l'entrée visuelle dans plusieurs canaux parallèles distincts. Navalpakkam et Itti [129] ont modélisé la recherche visuelle comme un problème d'optimisation de gain descendant en maximisant le rapport signal sur bruit (SNR) de la cible par rapport aux distracteurs au lieu d'apprendre des fonctions de fusion explicites. Autrement dit, ils ont appris des poids linéaires pour la combinaison de caractéristiques en maximisant le rapport entre la saillance de la cible et la saillance du distracteur. Autre modèles ont été proposés par [130] et [131].

- Les modèles bayésiens : combinent des preuves sensorielles avec des contraintes antérieures. Dans ces modèles, les connaissances antérieures (par exemple, le contexte de la scène ou l'essentiel) et les informations sensorielles (par exemple, les caractéristiques de la cible) sont combinées de manière probabiliste selon la règle de Bayes. Torralba [132] et Oliva et al. [133] ont proposé un framework bayésien pour les tâches de recherche visuelle. Dans la même direction, Ehinger et al. [134] ont intégré linéairement trois composantes (saillance ascendante, gist et objets) pour expliquer les mouvements oculaires dans la recherche de personnes dans une base de données d'environ 900 scènes naturelles. Itti et Baldi [135] ont défini des stimuli surprenants comme ceux qui modifient de manière significative les croyances d'un observateur. Ceci est modélisé dans un cadre bayésien en calculant la divergence entre les croyances postérieures et antérieures. Cette notion s'applique à la fois à l'espace (la surprise survient lorsque l'observateur affecte les croyances dérivées des localisations voisines) et au temps (la surprise survient lorsque l'observation des images à un moment donné affecte les croyances établies à partir d'observations antérieures).
- Les modèles théoriques de décision : L'interprétation de la théorie de la décision indique que les systèmes perceptuels évoluent pour produire des décisions sur les états de l'environnement qui sont optimaux dans un sens de décision théorique comme la probabilité d'erreur minimale. Le point essentiel est que l'attention visuelle doit être guidée par l'optimalité par rapport à la tâche finale. Gao et Vasconcelos [136] ont fait valoir que pour la reconnaissance, les caractéristiques saillantes sont ceux qui distinguent le mieux une classe d'intérêt de toutes les autres classes visuelles. Ils ont ensuite défini l'attention comme une classification avec une erreur minimale attendue. Après, Mahadevan et Vasconcelos [137] proposent une extension du modèle [136] avec un algorithme non supervisé pour la saillance spatio-temporelle basé sur les mécanismes biologiques du groupement perceptif basé sur le mouvement. La combinaison de la saillance du centre et de la puissance des textures dynamiques a rendu leur modèle applicable aux arrière-plans très dynamiques et aux caméras mobiles.

### 3.2.3.1 Le modèle Itti

Itti et al [5] proposent un système d'attention visuelle, inspiré par le comportement et l'architecture neuronale du système visuel précoce des primates. Le modèle proposé s'appuie sur une architecture biologique proposée par Koch et Ullman [138] et à la base de plusieurs modèles [139], [140]. Elle est liée à la théorie dite de «l'intégration des caractéristiques», proposée pour expliquer les stratégies de recherche visuelle humaine [141] [5].

D'abord, l'extraction des premières caractéristiques visuelles est appliquée pour décomposer l'image en un ensemble de maps caractéristiques. Avec  $r$ ,  $g$  et  $b$  étant les canaux rouges, vert et bleu de l'image d'entrée, une image d'intensité  $I$  est obtenue comme  $I = (r + g + b)/3$ .  $I$  est utilisé pour créer une pyramide gaussienne  $I(\delta)$ , où  $\delta \in [0..8]$  est l'échelle. Les canaux  $r$ ,  $g$  et  $b$  sont normalisés par  $I$  afin de découpler la teinte de l'intensité. Cependant, comme les variations de teinte ne sont pas perceptibles à très faible luminance (et ne sont donc pas saillantes), la normalisation n'est appliquée qu'aux endroits où  $I$  est supérieur à 1/10 de son maximum sur toute l'image. Quatre canaux de couleur largement accordés sont créés :

La map rouge

$$R' = r - \frac{(g + b)}{2} \quad (3.18)$$

la map verte

$$G' = g - \frac{(r + b)}{2} \quad (3.19)$$

La map bleue

$$B' = b - \frac{(r + g)}{2} \quad (3.20)$$

La map jaune

$$Y = \frac{r + g}{2} - \frac{|r - g|}{2} - b \quad (3.21)$$

Quatre pyramides gaussiennes  $R(\delta)$ ;  $G(\delta)$ ;  $B(\delta)$  Et  $Y(\delta)$ .

Les différences entre le centre  $c$  du masque et le voisinage  $s$  construisent les maps des caractéristiques. Le premier ensemble des maps concerne le contraste d'intensité qui, chez les mammifères, est détecté par des neurones sensibles soit à des centres sombres sur des environnements lumineux, soit à des centres lumineux sur des envi-

### Chapitre 3 : Localisation des panneaux routiers

ronnements sombres [5]. Ils utilisent une rectification pour calculer les deux types de sensibilités dans un ensemble de six maps  $I(c, s)$ , avec  $c \in [2, 3, 4]$  et  $s = c + \delta, \delta \in [2, 3, 4]$

Un second jeu de maps est construit de la même manière pour les canaux de couleur, qui sont représentés dans le cortex à l'aide d'un système dit de double-adversaire couleur : au centre de leur champ réceptif, les neurones sont excités par une couleur (par exemple rouge). Une telle opposition spatiale et chromatique existe pour les paires rouge/vert, vert/rouge, bleu/jaune et jaune/bleu dans le cortex visuel primaire humain [142]. En conséquence, les cartes  $RG(c, s)$  sont créées dans le modèle pour prendre en compte simultanément les doubles oppositions rouge/vert et vert/rouge, et  $BY(c, s)$  pour bleu/jaune et jaune/bleu double adversaire.

Au total, 42 maps de caractéristiques sont combinées dans la map de saillance, mais avant qu'ils ne soient combinés, ils doivent être normalisés. Le but de la map de saillance est de représenter la visibilité ou la saillance à chaque endroit du champ visuel par une quantité scalaire. Pour guider la sélection des emplacements fréquentés, en fonction de la distribution spatiale de la saillance. Une combinaison des cartes de caractéristiques fournit une entrée ascendante à la carte de saillance, modélisé comme un réseau de neurones dynamiques "winner-take-all". comme est illustré dans la figure 3.5.

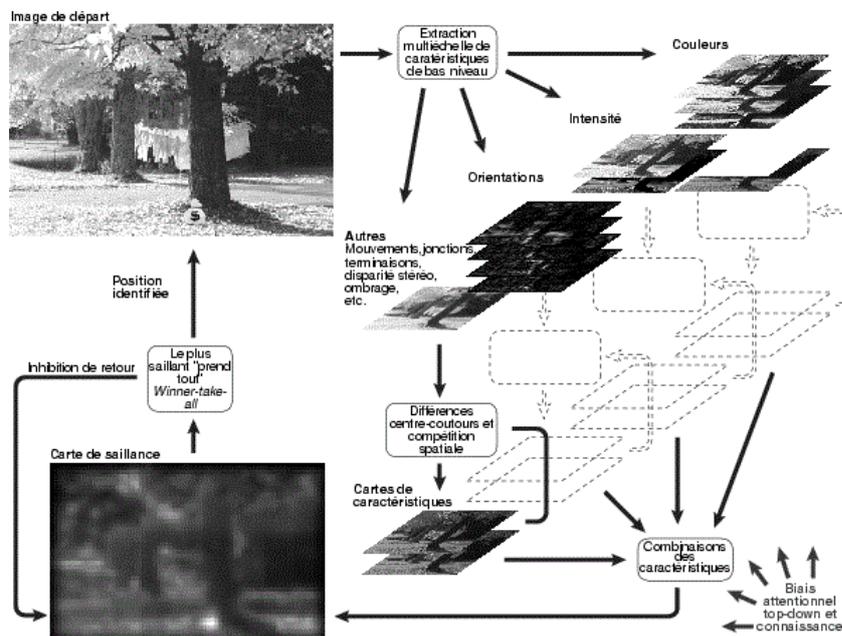


FIGURE 3.5 – Le modèle d'attention visuelle proposé par Itti et al [5].

Les modèles d'attention visuelle ont de nombreuses applications y compris la détection des panneaux routiers.

### 3.3 Contribution

Nous avons comparé plusieurs méthodes de segmentation afin de choisir la méthode qui répond le mieux à nos besoins. Premièrement nous avons testé l'espace HSV due à sa robustesse aux changements d'éclairage, mais nous avons trouvé que son coût en temps d'exécution et ses performances le rendent inutile. La figure 3.6 illustre les images converties en espace HSV.



FIGURE 3.6 – Les images converties en espace HSV, (a) image RGB, (b) image HSV.

Premièrement, nous avons testé la méthode de [6] présentée dans le chapitre 2 (equation 2.20). Avec cette méthode les panneaux sont mal localisés et nous trouvons un taux élevé de bruit comme est illustré dans la figure 3.7.

nous avons testé la méthode de [7] (voir equations 2.7,2.8 et 2.9 du chapitre 2) qui utilise un seuillage dans l'espace RGB, les résultats obtenus sont illustrés dans la figure 3.8. Cette méthode n'est pas robuste aux changements d'éclairage et beaucoup de panneaux rouges et bleus ne sont pas détectés due à la difficulté de trouver les valeurs optimales des seuils.

Pour éviter le problème de seuillage dans la segmentation nous avons testé les mé-

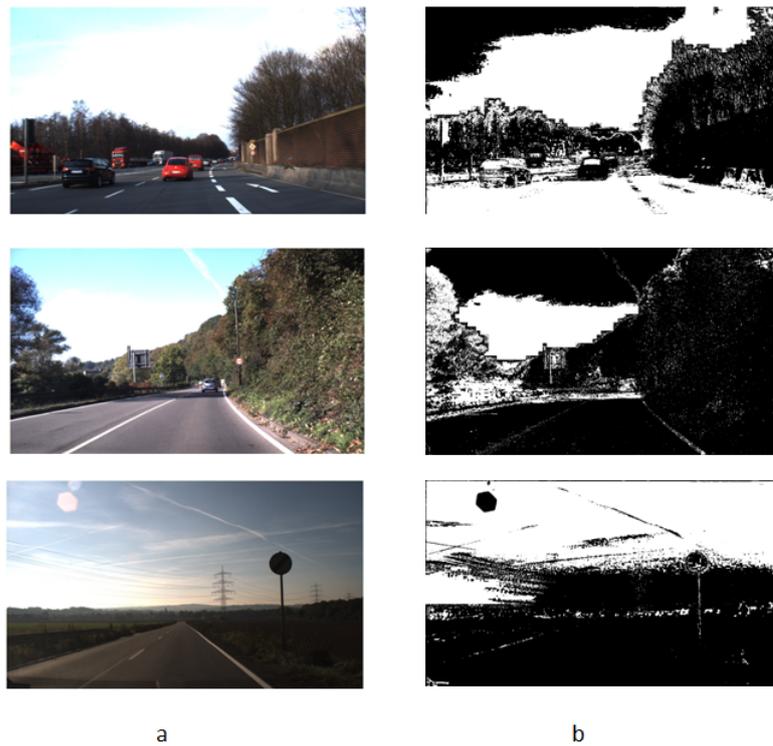


FIGURE 3.7 – Les images segmentées par la méthode de [6], (a) image RGB, (b) image segmentée.

thodes de [2] (voir les équations 2.4, 2.5 et 2.6 du chapitre 2 ), ainsi que la méthode de [8] qui utilise les équations du modèle itti [5] (les équations 3.18, 3.19, 3.20 et 3.21 du chapitre 2). La figure 3.9 illustre le résultat de la segmentation des deux méthodes.

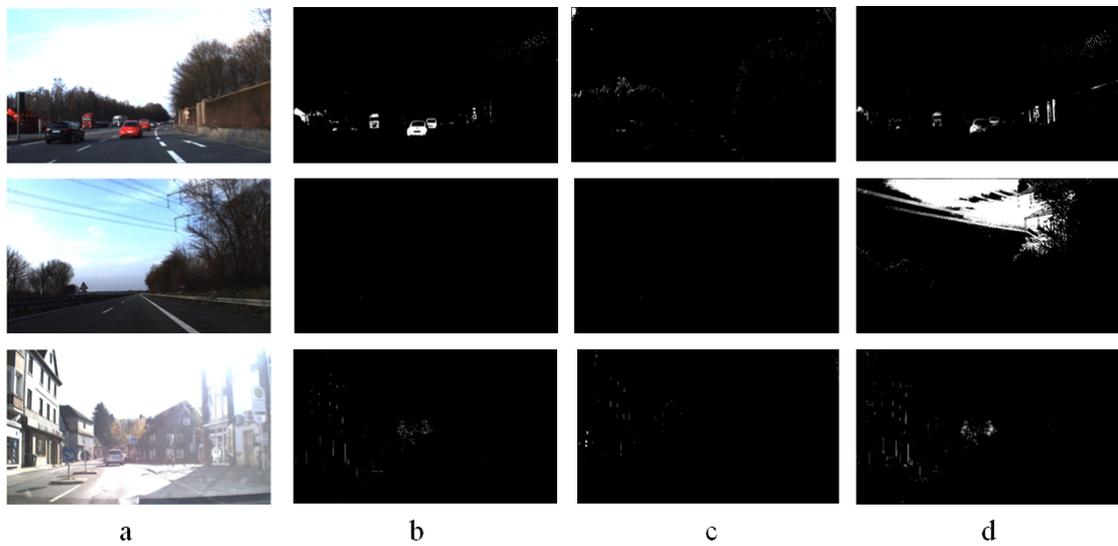


FIGURE 3.8 – Les images segmentées par la méthode de [7], (a) image RGB, (b) image rouge, (c) image bleue, (d) image jaune.

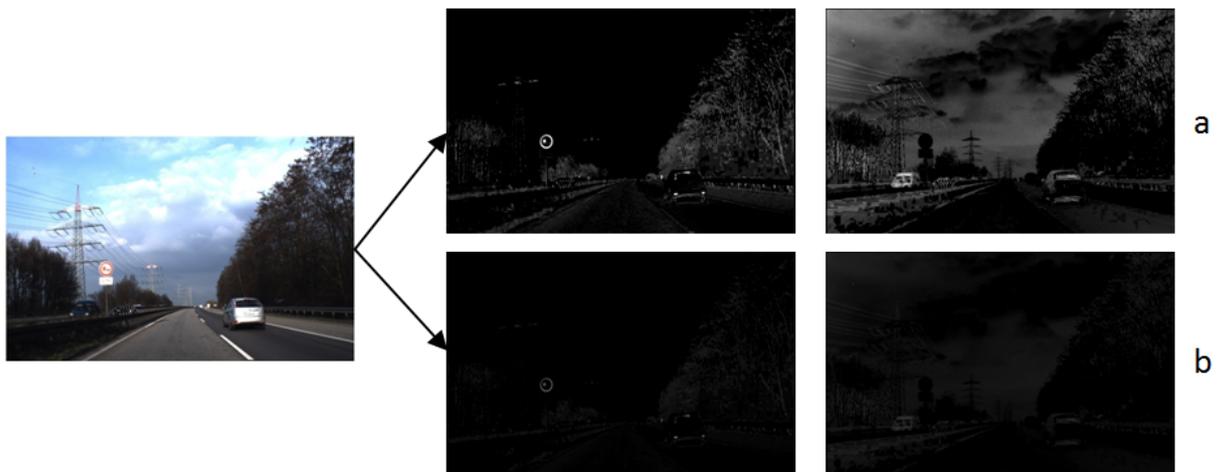


FIGURE 3.9 – Les images segmentées par la méthode de [2] (a) et la méthode de [8] (b)

Le but de la segmentation dans le domaine de la reconnaissance des panneaux routiers est de donner une haute intensité aux pixels qui ont une couleur dominante et d'éliminer autant que possible les pixels non désirés. Pour atteindre ces objectifs, nous proposons une nouvelle variante de la méthode proposée par [8] qui utilise les méthodes du modèle Itti, l'objectif est d'augmenter les intensités des régions d'intérêt et d'éliminer le maximum de pixels indésirables. L'idée est de multiplier le résultat de l'équation 3.18 par la valeur minimale entre la différence de canal rouge et canal bleu, et la différence de canal rouge et canal vert  $\min(r - g, r - b)$ . Le résultat de l'équation 3.20 est également multiplié par la différence minimale entre le canal bleu et les deux autres canaux  $\min(b - r, b - g)$  comme suit (les valeurs négatives sont mises à zéro et celles supérieures à 255 sont mises à 255) :

$$R = r - \frac{(g + b)}{2} \times \min(r - g, r - b) \quad (3.22)$$

$$B = b - \frac{(r + g)}{2} \times \min(b - r, b - g) \quad (3.23)$$

L'exemple suivant illustre l'avantage de la méthode proposée :

Si nous prenons un pixel  $x(26, 19, 6)$

$$I = 17, r = 1.52, g = 1.11, b = 0.35;$$

$$r - g = 0.41, r - b = 1.17$$

avec la méthode de lim et al [8]  $R=0.79$  et la valeur du pixel  $x$  dans la map rouge sera ( $0.79 \times 255 = 201$ ). Le pixel  $x$  n'est pas rouge, mais avec la méthode de lim et al [8] il est considéré comme un rouge, par contre avec notre méthode proposée la valeur du pixel  $x$  dans la map rouge sera ( $0.79 \times 0.41 \times 255 = 82$ ), donc l'intensité du pixel  $x$  se diminue avec la méthode proposée.

Ainsi, dans la map bleue avec un pixel  $x$  de valeur RGB (49, 57, 79) on obtient 107 avec la méthode de [8] mais avec notre méthode proposée cette valeur est réduit à 38.

Dans la figure 3.10 une comparaison entre la méthode de segmentation proposée et la méthode de lim et al [8]. Afin d'illustrer l'avantage de la méthode proposée nous avons appliqué la transformée de hough pour détecter les cercles dans les images segmentées, le nombre des cercles détectées est considérablement réduits comme est illustré dans l'image 3.11.



FIGURE 3.10 – Comparaison des images segmentées (a) image RGB, (b) méthode de lim et al [8], (c) notre méthode proposée.



FIGURE 3.11 – Comparaison des cercles détectés dans les images segmentées (a) avec notre méthode proposée, (b) avec la méthode de lim et al [8].

### 3.4 Conclusion

Dans ce chapitre nous avons présenté brièvement les méthodes de segmentation, les différents espaces de couleur et les modèles d'attention visuelle qui servent à détecter les régions saillantes ou homogènes dans les images. Nous avons choisi l'espace RGB normalisé car la conversion non linéaire et les performances de l'espace HSV le rendent inutile. Nous avons proposé une nouvelle méthode de segmentation inspirée du modèle Itti pour localiser les panneaux routiers dans la section 3.3 où l'objectif est d'augmenter les intensités des régions d'intérêt et d'éliminer le nombre maximum des pixels indésirables.

La transformée de hough confirme l'efficacité de la méthode proposée, où le nombre des cercles détectés est considérablement réduit par rapport aux cercles détectés dans les images segmentées avec la méthode de Itti.

## **Chapitre 4**

### **Détection des panneaux routiers**

### 4.1 Introduction

Après la localisation des régions d'intérêts contenant la couleur similaire à celle que l'on s'attend à trouver sur les panneaux de signalisation. Les panneaux routiers sont localisés avec la méthode de segmentation proposée dans le chapitre précédent, plusieurs objets ayant la même couleur seraient également détectés et considérés comme des segments positifs. C'est pour cela, nous utilisons la forme géométrique des panneaux pour éliminer les autres objets possèdent la même couleur que les panneaux mais pas la même forme géométrique. Par conséquent, l'élimination des faux positifs augmente la précision du système de reconnaissance des panneaux.

La détection des panneaux routiers, assume un rôle crucial dans toute application de reconnaissance des panneaux de signalisation routière. En fait, un signe qui n'est pas correctement détecté ne peut pas être classé et reconnu pour informer le conducteur. Par exemple, lorsque la zone de signalisation n'est pas complètement détectée, une mauvaise classification et une mauvaise reconnaissance sont susceptibles de se produire.

Dans ce chapitre, nous présentons d'abord dans la section 4.2.1 la transformé de hough utilisée pour la détection des panneaux circulaires , puis dans la section 4.2.2 nous détaillons LBP. Dans la section 4.3 nous présentons la nouvelle méthode proposée pour détecter les régions homogènes des images segmentées.

### 4.2 Background

#### 4.2.1 Transformé de hough

La méthode Hough Transform a été introduite, sous sa forme la plus élémentaire, par Hough en 1962, sous la forme d'un brevet [143]. Son application prévue était en physique des particules, pour la détection des lignes et des arcs dans les photographies obtenues dans les chambres de nuages. De nombreuses élaborations et améliorations de cette méthode ont été étudiées depuis.

La transformée de Hough classique est un algorithme standard pour la détection de ligne et de cercle. Il peut être appliqué à de nombreux problèmes de vision par ordinateur car la plupart des images contiennent des limites de caractéristiques qui peuvent être décrites par des courbes régulières. Le principal avantage de la technique de transformation de Hough est qu'elle tolère les lacunes dans les descriptions de limites d'entités et qu'elle est relativement peu affectée par le bruit de l'image, contrairement aux détecteurs de contours.

L'idée derrière la méthode est simple : les formes paramétriques dans une image sont détectées en recherchant des points d'accumulation dans l'espace des paramètres.

Si une forme particulière est présente dans l'image, alors le mappage de tous ses points dans l'espace des paramètres doit se regrouper autour des valeurs des paramètres qui correspondent à cette forme [144]. Cette approche cartographie les éléments distribués et disjoints de l'image en un point d'accumulation localisé, ce qui est à la fois un avantage et un inconvénient. Les formes qui s'occluent partiellement sont encore détectées, sur la preuve de leurs parties visibles. Par exemple, tous les segments d'un même cercle contribuent à la détection de ce cercle, quels que soient les écarts entre eux. D'un autre côté, les informations locales inhérentes aux points de la forme, comme la contiguïté, sont perdues, les extrémités des arcs de cercle et des segments de ligne doivent être déterminées dans une étape ultérieure.

La charge de calcul de la méthode augmente rapidement avec le nombre de paramètres qui définissent la forme détectée. Les lignes ont deux paramètres, les cercles trois et les ellipses ont cinq. La méthode de Hough a été appliquée à tous ceux-ci, mais l'ellipse est probablement à sa limite supérieure de praticité. Des tentatives ont également été faites pour appliquer la méthode de Hough à des formes arbitraires. Ces tentatives réduisent essentiellement la méthode à la correspondance de modèle, annulant les avantages qui découlent de l'analyse et quelques descriptions de paramètres de la forme [144].

Dans une image, une courbe "sous n'importe quelle forme" peut être définie par un ensemble de points. Il y a généralement un ensemble de paramètres qui relie ces points par leurs informations spatiales, ou coordonne des informations dans l'espace image. Cela signifie donc que la courbe est paramétrée et peut-être modélisée par une équation mathématique qui donne la relation entre deux ensembles (par exemple, l'équation d'une ligne, d'un cercle ...).

Ainsi, le principe général de la transformée de Hough est une projection de l'espace d'image  $N$ -dimensional (noté  $I$  et défini par sa variable  $i = 1 : N$ ) vers un espace de paramètre de dimension  $M$  (noté  $H$  et défini par sa variable  $j = 1 : M$ ). Les deux espaces sont liés par le modèle mathématique  $x_i = f(a_j)$ . L'objectif est donc de trouver les paramètres définissant la forme à détecter [144].

Par exemple, une droite est déterminée par l'équation :  $y = mx + p$ , où  $m$  représente la pente et  $p$  l'ordonnée à l'origine. Alors, la transformée de Hough est donnée par l'équation :

$$I = (x, y) \rightarrow H = (m, p) \tag{4.1}$$

Il existe deux types de la transformée de Hough (selon la formulation de l'équation mathématique reliant les deux espaces) :

- Transformation de 1 à  $m$  (un point de  $I$  est associé à  $m$  points de  $H$ ).
- Transformation de  $m$  en 1 (les  $m$ -points de  $I$  sont associés à un point de  $H$ ).

En pratique, l'espace de Hough  $H$  est discrétisé. Le résultat de la transformation de Hough est stocké dans une cellule de table de deux dimensions (dans le cas de l'espace  $(m, p)$  nommé l'accumulateur de Hough). Ainsi, chaque position de cellule  $m_i, p_i$ , possède son propre accumulateur.

### 4.2.1.1 Transformation de hough circulaire

La transformée de Hough peut être appliquée pour détecter la présence d'une forme circulaire dans une image donnée. Dans la transformation de hough circulaire, la procédure de vote est effectuée dans un espace de paramètres. Les maxima locaux dans l'espace accumulateur, obtenus par la procédure de vote, sont utilisés pour calculer la transformée de Hough. L'espace des paramètres est défini par la représentation paramétrique utilisée pour décrire les cercles dans le plan de l'image donnée par l'équation caractéristique d'un cercle de rayon  $r$  et centre  $(a, b)$  [144] :

$$(x - a)^2 + (y - b)^2 = r^2 \quad (4.2)$$

Ce cercle peut être décrit par les deux équations suivantes :

$$x = a + r \cos(\theta) \quad y = b + r \sin(\theta) \quad (4.3)$$

Ainsi, le rôle de la transformée de Hough est de rechercher le triplet de paramètres  $(a, b, r)$  qui détermine les points  $(x_i, y_i)$  comme décrit sur la figure 4.1.

Deux cas peuvent être présentés :

- Cas de rayon connu : si nous connaissons le rayon du cercle à détecter dans l'image, le paramètre à rechercher est réduit à une paire  $(a, b)$  et l'espace  $H$  est bi-dimensionnel. Nous considérons un cercle de rayon  $R$  et centre  $(a_0, b_0)$ , la transformation pour chaque point  $(x_i, y_j)$  dans l'espace  $I$  donne un cercle dans l'espace  $H$  ayant un centre  $(x_i, y_j)$  et un rayon  $R$ . De même, une transformation de tous les points du cercle en image est appliquée. Le résultat sera plus de cercles où leur intersection est le point  $(a_0, b_0)$ . Ce point est obtenu en cherchant le maximum de l'accumulateur [145].
- Cas de rayon non connu : dans ce cas, le travail consiste à trouver les paramètres du triplet  $(a_0, b_0, r_0)$  qui définissent les points de cercle à trouver, l'espace sera en 3D. Pour chaque point  $(x_i, y_j)$  de l'espace  $I$  correspondra à un cône dans l'espace  $H$ , comme le rayon  $r$  a varié de 0 à une valeur donnée. Après la transformation de tous les points de contour de la même manière, l'intersection donnera une surface sphérique correspondant au maxima de l'accumulateur. Caractérisé par un centre  $(a_0, b_0)$  et le rayon  $R$  recherché [145].

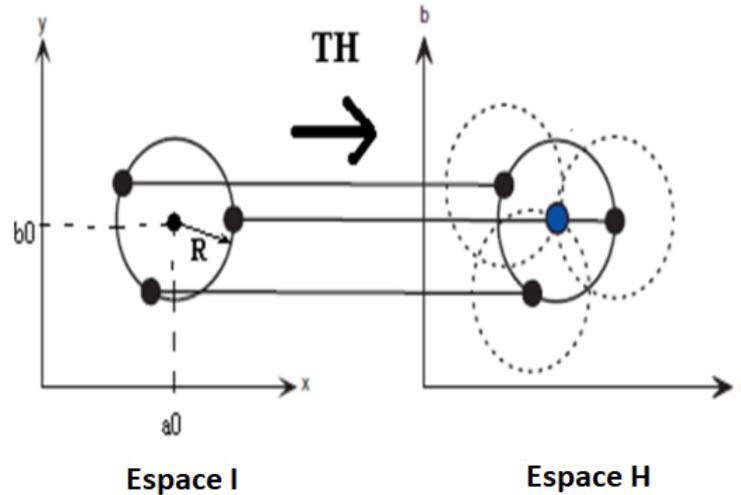


FIGURE 4.1 – Transformation d’un point dans un cercle.

#### 4.2.2 LBP

L’opérateur LBP (Local Binary Pattern) a été introduit en tant que mesure complémentaire du contraste de l’image locale [146]. il marque les pixels d’une image avec des nombres décimaux, appelés codes binaires locaux ou codes LBP, qui codent la structure locale autour de chaque pixel. Il procède ainsi, comme illustré dans la figure 4.2 : chaque pixel est comparé à ses  $p$  voisins dans un voisinage  $R$  en soustrayant la valeur de pixel central. Les valeurs strictement négatives sont codées avec 0 et les autres avec 1. Un nombre binaire est obtenu en concaténant tous ces codes binaires dans le sens des aiguilles d’une montre en commençant par celui en haut à gauche et la valeur décimale correspondante est utilisée pour l’étiquetage. Les nombres binaires dérivés sont appelés modèles binaires locaux ou codes LBP [147].

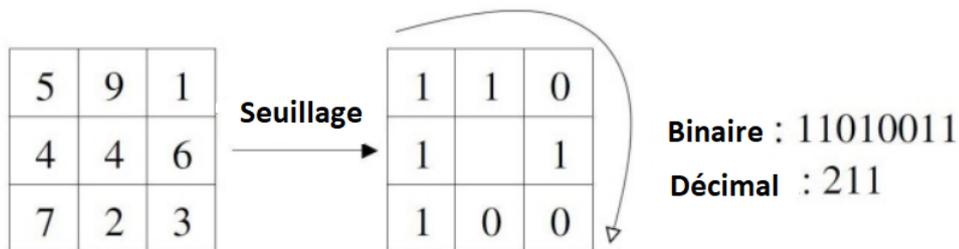


FIGURE 4.2 – Un exemple de LBP basique.

Une limitation de l'opérateur LBP de base est que son petit voisinage 3x3 ne peut pas capturer des caractéristiques dominantes avec des structures à grande échelle. Pour traiter la texture à différentes échelles, l'opérateur a ensuite été généralisé pour utiliser des voisinages de tailles différentes [148]. Un voisinage local est défini comme un ensemble de points d'échantillonnage régulièrement espacés sur un cercle centré sur le pixel à étiqueter, les points d'échantillonnage qui ne tombent pas dans les pixels sont interpolés par interpolation bilinéaire, permettant ainsi tout rayon et tout nombre de points d'échantillonnage dans le quartier. La figure 4.3 montre quelques exemples de l'opérateur LBP étendu, où la notation  $(P,R)$  indique un voisinage de  $P$  points d'échantillonnage sur un cercle de rayon  $R$ . Formellement, étant donné un pixel en  $(x_c, y_c)$ , le LBP résultant peut être exprimé sous forme décimale comme suit :

$$LBP_{P,R}(x_c, y_c) = \sum_{p=0}^{P-1} 2^p s(g_p - g_c) \quad (4.4)$$

Où  $g_c$  et  $g_p$  sont respectivement des valeurs de niveau de gris du pixel central et les pixels voisinages  $P$  de rayon  $R$ , la fonction  $s(x)$  est définie comme suit :

$$s(x) = \begin{cases} 1 & \text{si } x \geq 0 \\ 0 & \text{si } x < 0 \end{cases} \quad (4.5)$$

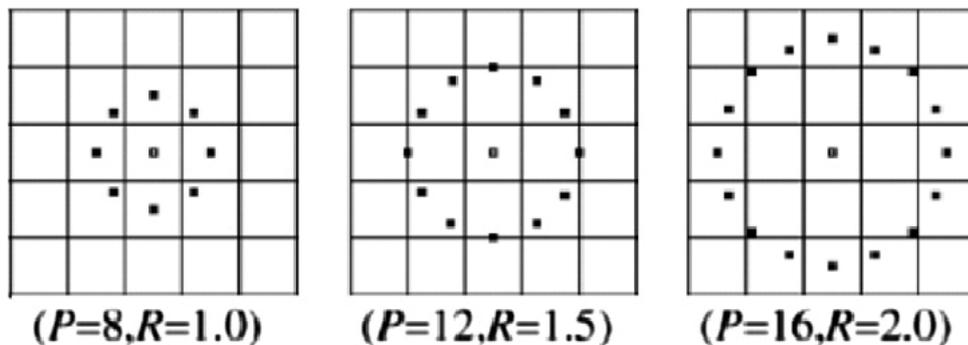


FIGURE 4.3 – Exemples de l'opérateur LBP étendu.

Par la définition ci-dessus, l'opérateur LBP de base est invariant par rapport aux transformations d'échelle de gris monotones préservant l'ordre d'intensité des pixels dans les voisinages locaux. L'histogramme des étiquettes LBP calculées sur une région peut être exploité comme un descripteur de texture [147].

L'opérateur  $LBP(P,R)$  produit  $2^P$  différentes valeurs de sortie, correspondant à  $2^P$  différents motifs binaires. Si l'image est pivotée, les pixels dans chaque voisinage  $p$  se

déplaceront de manière correspondante le long du périmètre du cercle, ce qui entraînera une valeur LBP différente, à l'exception des motifs avec seulement des valeurs 1 ou 0. Afin de supprimer l'effet de rotation, une variante de LBP robuste aux rotations est proposée dans [148] :

$$LBP_{P,R}^{r,i} = \min(ROR(LBP_{P,R}, i \mid i = 0, 1, \dots; p - 1)) \quad (4.6)$$

Où  $ROR(x, i)$  effectue un décalage  $i$  fois vers la droite du bit  $P$ . L'opérateur  $LBP_{P,R}^{r,i}$  quantifie les statistiques d'occurrence des modèles invariants de rotation individuel correspondant à certaines micro-caractéristiques de l'image. Par conséquent, les modèles peuvent être considérés comme un détecteur de caractéristiques [148]. Cependant, dans [149], il a été montré qu'un tel opérateur LBP invariant aux rotations ne fournit pas les informations discriminantes nécessaires, puisque les fréquences d'occurrence des motifs individuels incorporés dans  $LBP_{P,R}^{r,i}$  varient grandement et la quantification brute des espaces à des intervalles de  $45^\circ$ .

Il a été montré que certains modèles contiennent plus d'informations que d'autres [148]. Il est possible d'utiliser uniquement un sous-ensemble de motifs binaires  $2^P$  pour décrire la texture des images. Ojala et al. [148] a appelé ces motifs des motifs uniformes, notés  $LBP_{P,R}^{U2}$ . Un modèle binaire local est appelé uniforme s'il contient au plus deux transitions bit à bit de 0 à 1 ou inversement lorsque la chaîne de bits correspondante est considérée comme circulaire. Par exemple, 00000000 (0 transitions) et 01110000 (2 transitions) sont tous deux uniformes alors que 11001001 (4 transitions) et 01010011 (6 transitions) ne le sont pas. Les motifs uniformes représentent environ 90% de tous les motifs dans un voisinage (8,1) et environ 70% dans un voisinage (16,2) dans des images de texture [148]. Une expérience similaire a été menée sur la base de données FERET, et il a été constaté que 90,6% des motifs dans un voisinage (8,1) et 85,2% dans un voisinage (8,2) sont uniformes [150]. Shan et Gritti [151] ont vérifié la validité des motifs uniformes pour représenter les visages du point de vue de l'apprentissage automatique. Spécifiquement, ils ont appliqué AdaBoost pour sélectionner les modèles discriminants pour la reconnaissance d'expression faciale, leurs expériences ont démontré que, en utilisant l'opérateur LBP (8, 2), 91,1% de ces modèles sélectionnés sont uniformes. Accumuler les motifs non uniformes dans un seul casier donne un opérateur LBP avec moins de  $2^P$  étiquettes. Par exemple, le nombre d'étiquettes avec le voisinage de 8 pixels est de 256 pour la LBP standard mais seulement de 59 pour  $LBP^{U2}$ .

Il convient de noter qu'à peu près à la même époque où l'opérateur LBP a été proposé, Zabih et Woodfill [152] ont introduit une méthode de transformation CT (Census Transform) qui est très similaire à la lombalgie. CT fait également correspondre le voisinage local entourant un pixel sur une chaîne binaire, la seule différence entre LBP et CT est l'ordre inverse de la chaîne de bits. Plus tard, CT et ses variations ont été exploitées pour l'analyse d'images faciales [147].

### 4.3 Contribution

Dans le but de détecter les panneaux routiers et reconnaître leurs formes géométriques, nous avons testé deux méthodes : la détection des contours et la détection des régions homogènes ayant une forte intensité.

Dans un premier temps, nous avons binarisé l'image segmentée avec la méthode otsu puis détecter les contours de l'image binarisée en utilisant le filtre de sobel ou le filtre canny, après un processus de reconnaissance de forme géométrique est appliqué comme nous l'avons montré dans [4], mais le problème est qu'il y a des coupures dans les contours détectés causé par le bruit dans l'image ce qui réduit les performances de la méthode proposée pour la reconnaissance des formes géométrique. Ainsi que le nombre important des faux positifs détectés. Comme première solution nous avons filtré l'image segmentée avec le filtre gaussien afin de réduire le bruit, car avec le filtre gaussien plus un signal est lisse moins le lissage aura d'effet sur celui-ci. Donc, que les zones non homogènes seront affectées par ce filtre, mais dans notre cas ça ne donne pas de bons résultats à cause de la complexité des images.

Si nous observons les images segmentées dans le chapitre 3, nous remarquons que les régions où se trouve un panneau sont homogène par contre le reste du bruit est une zone non homogène. Premièrement, nous avons proposé d'utiliser la variance locale pour détecter les régions homogènes, et remplacer la valeur du pixels par sa variance avec ses voisins mais le résultat est indésirable. Comme deuxième proposition, nous divisons la valeur du pixel de l'image en entrée par la variance de la région du rayon  $R$ . plus la région est homogène plus la variance est minimale. Donc, la division de la valeur des pixels de cette région par la variance va augmenter leurs intensités. Par contre, si la région est non homogène sa variance sera grande et la la division de ses pixels par la variance va diminuer leurs intensités. L'algorithme suivant détaille le principe de cette idée.

---

**Algorithm 1** Calculer image variance

---

**Require:**  $img$  : image de taille  $(H, W)$ ,  $\omega$  : taille du masque

**for**  $i=0$  to  $H$  **do**

**for**  $j=0$  to  $W$  **do**

$var = variance(img, i, j, \omega)$  (Calculer la variance pour le mask de taille  $\omega$ )

$image_{var}(i, j) = (img(i, j) / var) * 255$ .

**end for**

**end for**

---

L'algorithme proposé diminue énormément les régions indésirables, mais encore élimine beaucoup de panneaux routiers ce qui diminue le rappel du système ainsi que le temps d'exécution de cet algorithme (3 sec/image) le rend inutile.

Une comparaison entre les deux méthodes qui utilisent la variance pour filtrer l'image est montrée dans la figure 4.4.

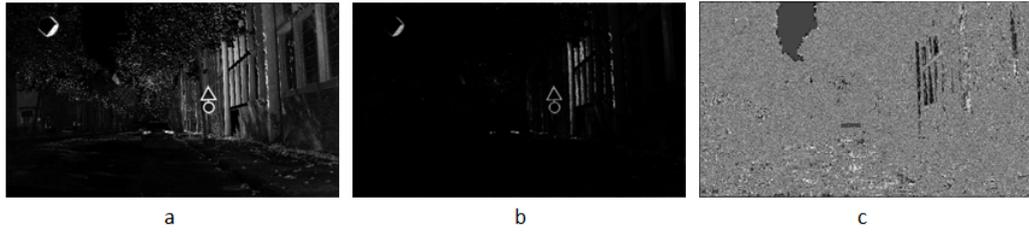


FIGURE 4.4 – Comparaison des images segmentées (a) image segmentée, (b) image filtrée avec la méthode proposée, (c) image filtré avec la variance.

Une deuxième méthode basée sur l'entropie pour la détection des régions homogènes (salient regions) est proposée. L'entropie d'une image est un indicateur de sa complexité, si l'image est uniforme et ne possède qu'une couleur, son entropie est nulle. Plus l'entropie est élevée, plus le contenu de l'image est aléatoire. Donc les régions homogènes possèdent une entropie proche de 0. pour détecter les régions homogènes, le même principe de la fenêtre glissante est utilisé. Mais le temps de calcul de l'entropie est long à cause de l'utilisation du logarithme ce qui nous empêche d'utiliser cette méthode.

Le LBP est largement utilisé dans l'extraction des caractéristiques pour sa rapidité et son efficacité. Nous avons proposé une nouvelle variante basée sur LBP et l'utilisé comme un filtre pour détecter les régions homogènes. L'idée est de fixer le poids  $p$  de l'équation 4.4, autrement dit donner le même poids pour toutes les fonctions  $s(x)$  comme est illustré dans l'équation suivante

$$LBP_{P,R}(x_c, y_c) = \sum_{p=0}^{P-1} 2^n s(g_p - g_c) - 1 \quad (4.7)$$

Comme  $p = 8$  et  $R = 1$ , pour obtenir la valeur maximale (255) dans le pixel  $(x_c, y_c)$  on met  $n = 5$  donc  $(8 \times (2^5) = 256)$ , on obtient l'équation suivante :

$$LBP_{p,R}(x_c, y_c) = \sum_{p=0}^{P-1} 2^5 s(g_p - g_c) - 1 \quad (4.8)$$

Pour calculer des régions homogènes nous considérons seulement les pixels similaires avec des intensités proches selon un seuil  $th$ . C'est pourquoi nous calculons la valeur absolue de  $(g_p - g_c)$  car l'objectif est de trouver les pixels homogènes. Nous dé-

finissons la fonction  $s(x)$  comme suit :

$$s(x) = \begin{cases} 1 & \text{si } |x| \geq th \\ 0 & \text{si } |x| < th \end{cases} \quad (4.9)$$

L'avantage de la méthode proposée est illustré dans la figure 4.5 où nous fixons expérimentalement  $th = 8$ .

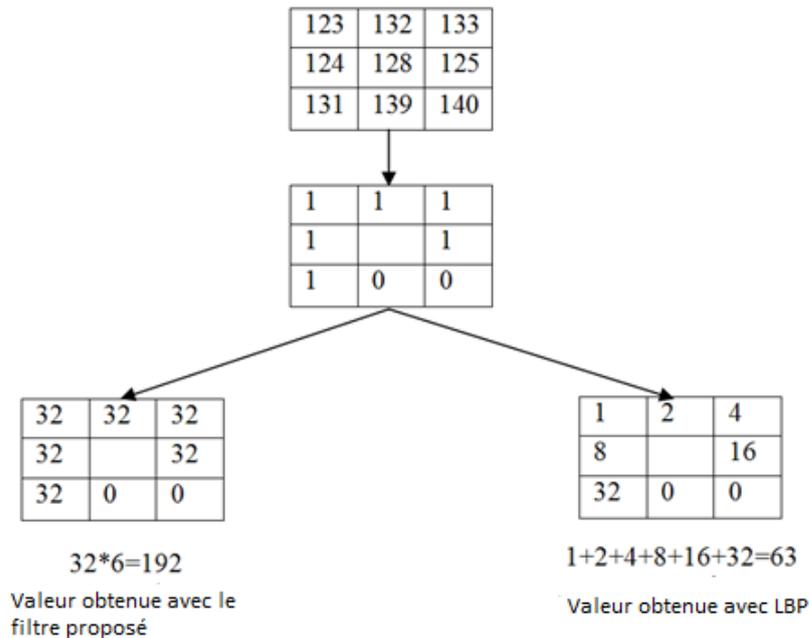


FIGURE 4.5 – Exemple de la méthode proposée basée sur LBP.

Comme l'objectif est de détecter les régions homogènes, nous utilisons le LBP comme un filtre et non comme un extracteur de caractéristiques. Si nous observons l'exemple nous remarquons que les pixels de la région sont similaires, avec la méthode proposée basée sur LBP nous obtenons une haute intensité du pixel  $(x_c, y_c)$ , cependant, avec le LBP classique nous avons une faible intensité et ne sera pas considéré comme un pixel d'une région homogène.

Pour les panneaux circulaires nous avons utilisé la transformé de hough vue sa rapidité et sa robustesse aux bruits, dans l'image 4.6 nous présentons une comparaison entre les cercles détectés dans les images segmentées avec et sans la méthode de filtrage proposée basée sur LBP. Le filtrage appliquée permet de réduire le nombre des fausses alarmes détectées dans l'image 4.6 ce qui permet d'augmenter la précision du système.

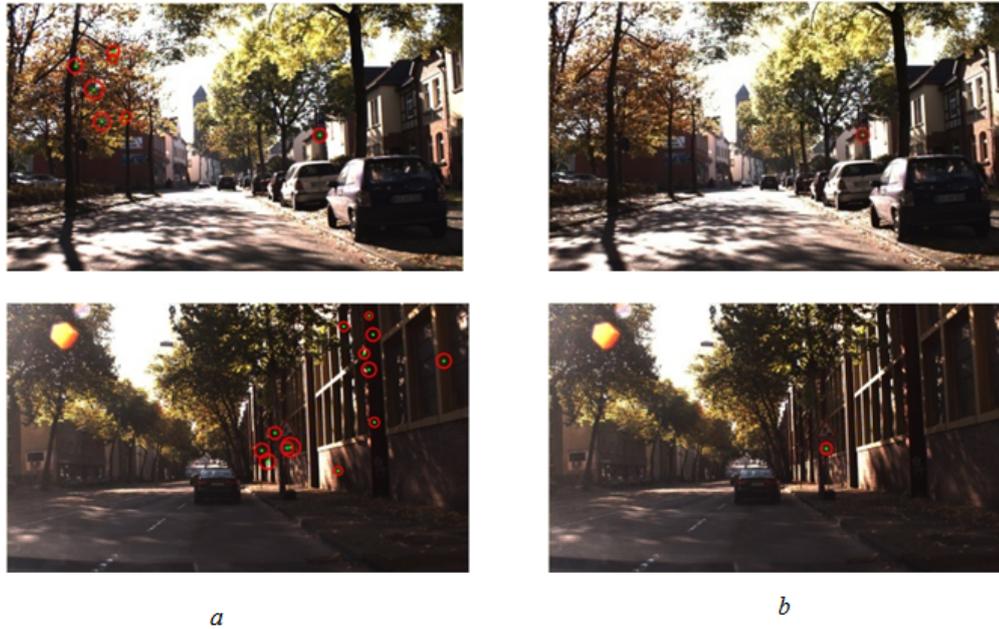


FIGURE 4.6 – Effet de la méthode proposée basée sur LBP, (a) sans la méthode proposée, (b) avec la méthode proposée.

### 4.4 Conclusion

Dans ce chapitre nous avons présenté brièvement la théorie de la transformée de hough et LBP. Pour détecter les panneaux routiers, comme première solution, nous avons binarisé l'image segmentée avec la méthode otsu puis détecter les contours avec le filtre de sobel ou le filtre Canny. Cependant, le problème est que les contours détectés ne sont pas tous fermés et il y a une coupure dans les contours. Ainsi que le nombre des faux positifs détectés est très élevé. Même avec un filtre gaussien ça ne donne pas de bons résultats. Nous avons proposé un nouveau filtre basé sur LBP pour détecter les régions homogènes où nous avons fixé le même poids pour toutes les différences  $g_p - g_c$ . Pour détecter les régions homogènes nous calculons la valeur absolue de  $g_p - g_c$  et si cette valeur est  $> th$  donc  $S(g_p - g_c) = 1$  sinon c'est 0. Le nombre des cercles détectés avec la transformée de hough est considérablement réduit en appliquant le filtre proposé.

## **Chapitre 5**

# **Reconnaissance des panneaux routiers**

## 5.1 Introduction

La reconnaissance d'un panneau routier dépend de son contenu enfermé dans une forme géométrique, le contenu du panneau routier varie en fonction du message qu'il est destiné à transmettre aux usagers de la route. Dans ce chapitre, le contenu interne d'un panneau routier est traité de manière à être distingué et identifié des autres classes de la signalisation routière.

Le contenu interne d'un panneau de signalisation ou le pictogramme contient les significations qui distinguent les panneaux de signalisation qui ont la même couleur et la même forme géométrique, par ex. des panneaux définissant des limites de vitesse de 30 et 50. Le pictogramme est normalement situé au centre d'un panneau routier bordé par une couleur et une forme particulières.

Dans ce chapitre nous présentons les méthodes utilisées dans la reconnaissance des panneaux routiers, nous comparons différentes méthodes de points d'intérêt, nous testons les performances de template matching et SVM. Nous présentons aussi l'histogramme des gradients orientés utilisés comme caractéristique avec le SVM.

## 5.2 Détection et description des points d'intérêt.

Les points d'intérêt sont des endroits saillants, faciles à repérer, ils correspondent à une discontinuité des niveaux de gris et, selon ces changements d'intensité on peut trouver différents types de points d'intérêt : des coins simples ou des jonctions en "V", "T", "L", ou encore en "damier" comme est schématisé dans la figure 5.1. Nous présentons dans ce qui suit deux méthodes d'indexation d'image basées sur les points d'intérêt robuste aux rotations et aux changement d'échelle à savoir SIFT (Scale Invariant Feature Transform) [9] et SURF (Speed Up Robust Features)[153].

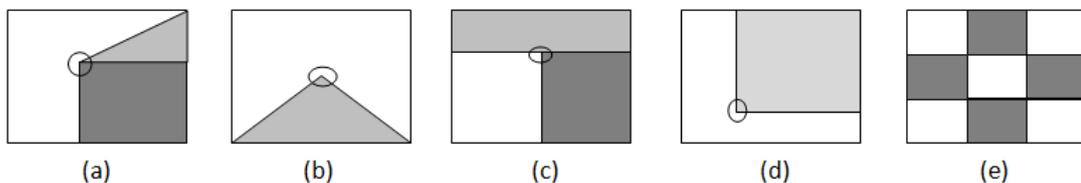


FIGURE 5.1 – Différents types de points d'intérêt : (a) coin simple, (b) jonction en "V", (c) jonction en "T", (d) jonction en "L", (e) jonction en "damier".

### 5.2.1 Descripteur SIFT

Le descripteur SIFT proposé dans [9] combine les DoG (Difference of Gaussian) qui sont invariants aux translations, rotation et mise à l'échelle avec un descripteur basé sur les distributions d'orientations de gradient qui est de plus robuste aux changements d'illumination et les différents points de vues. Les étapes principales pour calculer ce descripteur sont :

- **Détection d'extrema dans l'espace-échelle gaussien** : La première étape du calcul consiste à rechercher toutes les échelles et tous les emplacements d'image. Il est mis en oeuvre efficacement via le calcul des extrema des différences de gaussienne convoluées avec l'image pour identifier les points d'intérêt potentiels invariants à l'échelle et à l'orientation. L'espace-échelle est défini comme la fonction  $L(x, y, \sigma)$ , produit de convolution entre la gaussienne  $G(x, y, \sigma)$  et l'image  $I(x, y)$

$$L(x, y, \sigma) = G(x, y, \sigma) \otimes I(x, y) \quad (5.1)$$

Avec  $\otimes$  est le produit de convolution et

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2} \quad (5.2)$$

La fonction différence de gaussienne convoluée avec l'image s'écrit alors :

$$D(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma) \quad (5.3)$$

Où  $k$  un facteur multiplicateur constant généralement  $k = \sqrt{2}$ .

Cette approche gère la notion d'espace d'échelle, par le biais des  $n$  octaves, à l'intérieur desquelles l'image est convoluée par une gaussienne d'écart type  $k\sigma$ , autrement dit une octave correspond au doublement de l'échelle du flou gaussien  $\sigma$ . Les images résultantes des différences de gaussiennes permettent de déterminer les maxima locaux comme est illustré dans la figure 5.2 [154]. Pour chaque octave d'espace d'échelle, l'image initiale est convoluée plusieurs fois avec les Gaussiens pour produire l'ensemble des images d'espace d'échelle présentées à gauche. Les images gaussiennes adjacentes sont soustraites pour produire les images de différence de gaussiennes à droite. Après chaque octave, l'image gaussienne est sous-échantillonnée par un facteur de 2 et le processus est répété. Une octave correspond au doublement de l'échelle du flou gaussien  $\sigma$ .

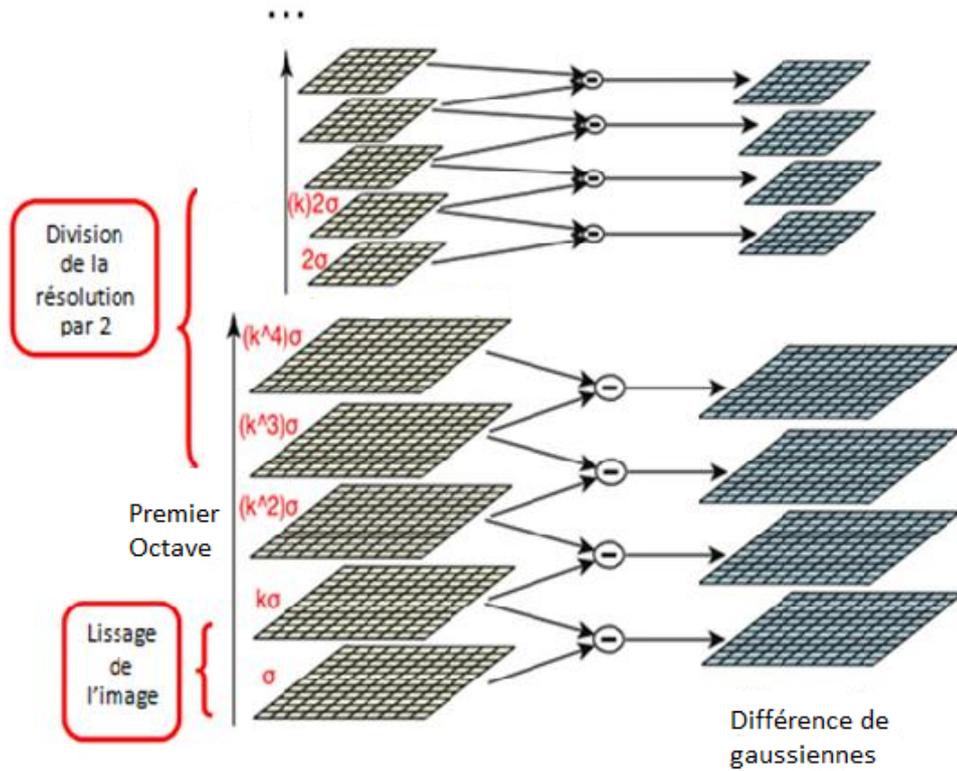


FIGURE 5.2 – Détection d'extrema dans l'espace-échelle gaussien.

- **Localisation des points d'intérêt** : En effet, un point est dit d'intérêt si sa valeur est maximale ou minimale dans l'ensemble de ses 26 connexités dans la même octave (huit voisins dans l'image actuelle et ses neuf voisins dans l'échelle ci-dessus et neuf dans l'échelle ci-dessous) comme est illustré dans la figure 5.3. Les points d'intérêt candidats correspondant à une zone de faible contraste, ou ceux dont la réponse est analogue à un contour sont éliminés.

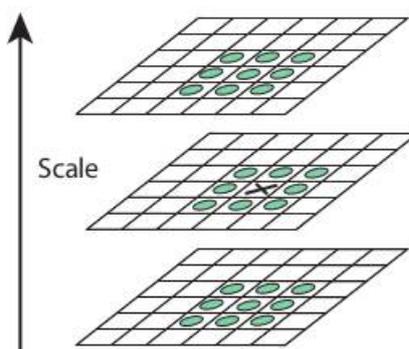


FIGURE 5.3 – Espace d'échelle gaussien. On recherche les extrema des différences de gaussienne afin de déterminer la position des points d'intérêt.

- **Calcul du descripteur** : Une fois qu'un point d'intérêt a été obtenu et est muni d'une position et d'une échelle, il est possible de calculer son descripteur. Tout d'abord, pour chaque pixel  $(x,y)$  un histogramme des orientations locales est calculé :

$$\theta(x, y) = \arctan\left(\frac{L(x, y + 1) - L(x, y - 1)}{L(x + 1, y) - L(x - 1, y)}\right) \quad (5.4)$$

Cet histogramme se compose de trente-six intervalles (36 classes), couvrant chacun un angle de dix degrés. Ce dernier est pondéré d'une part par un filtre gaussien d'écart-type égal à 1.5 la valeur de l'échelle locale et d'autre part par l'amplitude  $m$  de chaque point défini par :

$$m(x, y) = \sqrt{(L(x + 1, y) - L(x - 1, y))^2 - (L(x, y + 1) - L(x, y - 1))^2} \quad (5.5)$$

Les pics ainsi obtenus correspondent aux orientations dominantes. La fenêtre de description du voisinage du point d'intérêt possède une taille fixe de 16x16 pixels, subdivisée en un mask de taille 4x4 pixels chacune. On ré-échantillonne ensuite par interpolation les orientations des gradients suivant 8 directions principales, en groupant des carrés de 4 pixels de côté. On obtient ainsi pour une zone de taille 4x4 pixels avec 8 orientations un descripteur contenant 128 éléments (4x4x8). La figure 5.4 représente un descripteur sur une grille de taille 2x2, i.e. contenant 4 histogrammes statistiques sur les directions du gradient dans ce voisinage [154].

L'inconvénient majeur de SIFT est son temps d'exécution, et la grande taille de son descripteur.

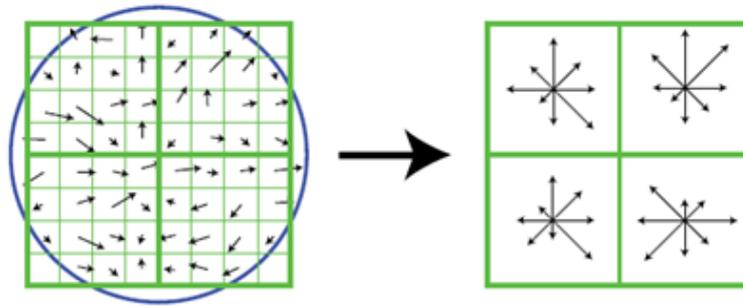


FIGURE 5.4 – À gauche, détails des valeurs et directions des gradients; à droite, descripteur SIFT correspondant. Le cercle bleu matérialise la pondération gaussienne suivant l'éloignement du centre de la zone clé [9].

### 5.3 Descripteur SURF

L'algorithme SURF [153] est composé de deux étapes principales : la première consiste à détecter des points d'intérêt sur l'image, la seconde calcule un descripteur de 64 caractéristiques pour chaque points d'intérêt.

- **Détection des points d'intérêt** : Bay et al [153] utilisent l'image intégrale afin de gagner en temps de calcul. Soit  $i$ , notre image de départ,  $i(x, y)$  représente la valeur d'un pixel de l'image aux coordonnées  $x$  et  $y$ . L'image intégrale, notée  $ii(x, y)$ , est une image de même taille que l'image d'origine, calculée à partir de celle-ci. Chaque pixel de l'image intégrale contient la somme des pixels situés au dessus et à gauche de ce pixel dans l'image initiale. La valeur d'un pixel de l'image intégrale  $ii$  est définie à partir de l'image  $i$  par l'équation suivante :

$$ii(x, y) = \sum_{x'=0}^{x \leq x} \sum_{y'=0}^{y' \leq y} i(x', y') \quad (5.6)$$

Sur la base de l'image intégrale, seules trois opérations (addition ou soustraction) sont nécessaires pour calculer la somme des valeurs d'intensité des pixels sur une zone rectangulaire de l'image.

Pour la détection des point d'intérêt, la matrice FastHessian est utilisée. Le déterminant de la matrice est utilisé pour décider si un point peut être choisi comme point d'intérêt ou non. Dans une image  $I$ , la matrice de Hesse au point  $X$  et à l'échelle de  $\sigma$  est définie comme suit :

$$H(x, \sigma) = \begin{bmatrix} L_{xx}(X, \sigma) & L_{xy}(X, \sigma) \\ L_{yx}(X, \sigma) & L_{yy}(X, \sigma) \end{bmatrix} \quad (5.7)$$

Où  $L_{xx}(X, \sigma)$  (laplacien de gaussien) est la convolution de la dérivée gaussienne du second ordre avec l'image  $I$  au point  $X$  et de manière similaire pour  $L_{xy}(X, \sigma)$  et  $L_{yy}(X, \sigma)$ .

Les zones de fort changement d'intensité des pixels sont recherchées dans l'image. La matrice Hessienne  $H$ , basée sur le calcul des dérivées partielles d'ordre deux, est utilisée pour cela. On définit un point critique pouvant être, soit dégénéré lorsque le hessien (déterminant de la matrice hessienne) s'annule, soit non dégénéré et dans ce cas, il faut étudier sa nature (point d'extremum local ou point selle) à travers le signe des valeurs propres de la matrice  $H$ .

- Si ces dernières sont positives, le point constitue un minimum local;
- Si elles sont négatives, il constitue un maximum local;
- S'il y a des valeurs propres de chaque signe, le point définit un point selle (point col).

Les points d'intérêt seront donc localisés là où le déterminant de la matrice Hessienne est maximal. Concrètement, les dérivées partielles du signal sont calculées par un produit de convolution avec des gaussiennes. Afin de gagner en rapidité, ces gaussiennes sont approximées par une fonction à paillier appelée box filter.

Pour gérer le multi-échelle, Bay et al s'appuient sur un ensemble de masques de tailles croissantes ( $9 \times 9$ ;  $15 \times 15$ ;  $21 \times 21$ ; ...) dépendant de l'écart type de la gaussienne à approximer [10]. Au final, les points d'intérêt dont le déterminant de la matrice hessienne est positif et qui sont maximums locaux dans un voisinage  $3 \times 3 \times 3$  (abscisse  $\times$  ordonnée  $\times$  échelle) sont conservés [154].

- **Description des points d'intérêt** : Après la détection des points d'intérêt, la deuxième étape consiste à calculer le descripteur correspondant. Le descripteur SURF est fortement influencé par l'approche de SIFT, il couple une étape de recalage de la zone d'analyse avec la construction d'un histogramme de gradients orientés. Pour déterminer l'angle de recalage bay et al [153] appliquent les ondelettes de Haar sur l'image intégrale pour calculer les dérivées premières de l'image sur un voisinage carré et d'étudier ainsi la répartition des gradients horizontaux et verticaux, ce qui permet de diminuer significativement le temps de calcul. La réponse en  $x$  et en  $y$  des ondelettes de Haar est calculée dans un voisinage de  $6s$  où  $s$  est l'échelle à laquelle le point d'intérêt a été trouvé. A partir de ces valeurs, l'orientation dominante de chaque point d'intérêt est calculée en faisant glisser une fenêtre d'orientation comme est schématisé dans la figure 5.5.

Le calcul du descripteur se base sur les sommes des réponses des ondelettes horizontales et verticales ainsi que sur leurs normes, tout d'abord le masque d'analyse du SURF de taille 20 centré sur le point d'intérêt et orienté selon l'orientation dominante extraite. Cette zone est subdivisée en  $4 \times 4$  carrés. Pour chacune de ces sous-régions, les ondelettes de Haar sont calculées sur une région de  $5 \times 5$  points. Soit  $d_x$  et  $d_y$  la réponse à l'ondelette de Haar, 4 valeurs sont calculées pour chacune des sous-régions :

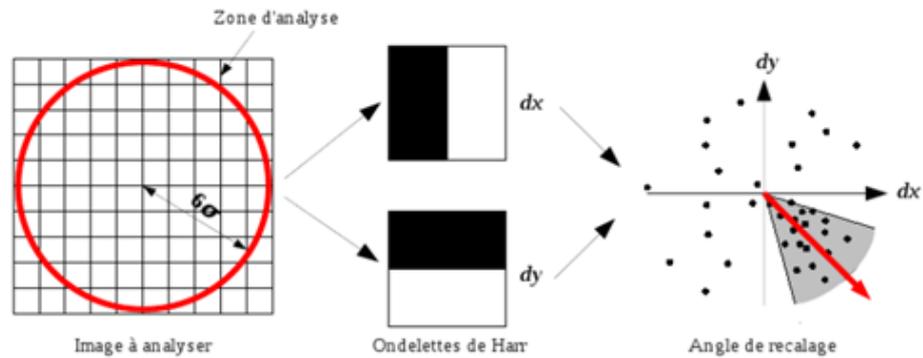


FIGURE 5.5 – Détermination de l’angle de recalage du SURF, en analysant la répartition des réponses des ondelettes de Haar [10].

$\Sigma d_x, \Sigma d_y, \Sigma |d_x|, \Sigma |d_y|$  comme est illustré dans la figure 5.6. Au final, chacun des points extraits est décrit par un vecteur composé de  $4 \times 4 \times 4$  valeurs soit 64 dimensions.

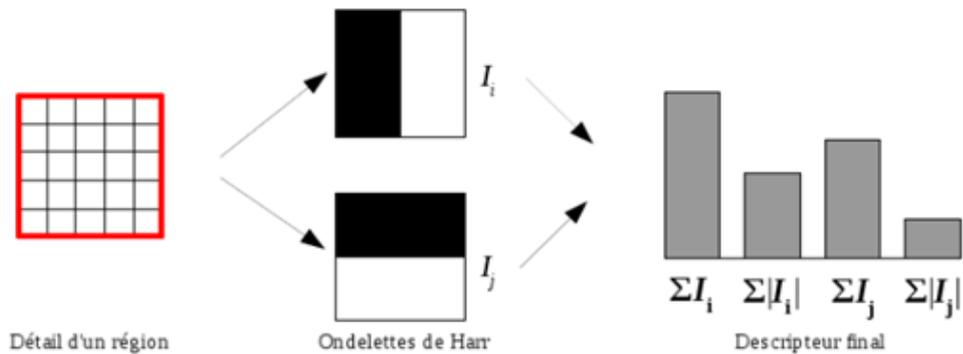


FIGURE 5.6 – Extraction des différentes composantes du descripteur SURF par le biais des ondelettes de Haar [10].

### 5.3.1 Appariement

L’appariement sert à diminuer le gap sémantique entre le bas niveau (extraction et description) et le haut niveau (reconnaissance) tout en cherchant le couple de points ayant un taux élevé de similarité. Dans l’appariement, chaque point de l’image requête est associé au point le plus proche de l’image template. La recherche Plus Proche Voisin (PPV) peut s’avérer longue mais, dans le cas des panneaux routiers il n’y a qu’une

dizaine de points d'intérêt. L'algorithme de FLANN décrit dans [155] utilise le principe d'arbres KD aléatoires proposé par Slipa et al [156]. Les arbres KD permettent de structurer l'espace de recherche afin d'accélérer la comparaison d'un élément avec les autres, mais les performances de recherche d'un arbre KD se rapprochent de celles d'une recherche linéaire lorsque la dimension de l'espace de données est grande. Un arbre KD ne serait donc pas performant pour SURF puisqu'il y a 64 dimensions. Dans notre travail nous avons proposé d'utiliser le descripteur SURF pour la reconnaissance des panneaux routiers, pour chaque image ROI détectée dans la phase de détection (4), nous appliquons un appariement entre l'image détectée et toutes les images de la même classe.

Le problème ce n'est que de trouver un bon appariement, mais aussi de mesurer cet appariement pour choisir l'image la plus proche, c'est pour cela, on a essayé premièrement de calculer une corrélation entre les descripteurs des points appariés des deux images, puis choisir le couple ayant une corrélation maximale, mais ça ne donne pas de bons résultats. Une deuxième méthode sert à calculer un coefficient  $\sigma$  défini comme suit :

$$\sigma = \frac{\text{nombre de bons appariements}}{\text{nombre total d'appariements}} \quad (5.8)$$

Le nombre des bons appariements est calculé en fixant une distance maximale entre les descripteurs, l'image la plus proche est celle qui possède un coefficient  $\sigma$  élevé. Mais ce nombre ne donne aucune information sur les distances entre les descripteurs pour mesurer le taux de similarité. C'est pour cela dans [4] nous préférons de combiner le coefficient avec la moyenne des distances comme suit :

$$dist = \frac{1}{N} \times \sum_{i=1}^N dist(p_i, q_i) \times \sigma \quad (5.9)$$

L'image la plus proche est celle qui a une distance  $dist$  minimale, un exemple du résultat d'appariement est illustré dans la figure 5.7.

L'inconvénient de cette méthode est que l'appariement se fait pour chaque image ROI détectée et tous les panneaux de la même classe, cela nécessite de nombreux calculs. Pour chaque panneaux le temps d'appariement est 50 ms, et cela reste loin d'un système temps réel. Le deuxième inconvénient est que la méthode n'est pas robuste aux changement de luminosité.



FIGURE 5.7 – Appariement des images requête avec l'image la plus proche.

### 5.4 Correspondance de modèle (Template matching)

Template Matching est une méthode de vision artificielle de haut niveau qui détermine les composants d'une figure qui correspond à un modèle prédéfini. Template Matching pourrait être une méthodologie dans le traitement d'image numérique pour identifier les petits composants d'une figure qui correspondent à une image modèle. Faire correspondre un modèle à une image quel que soit le modèle est une sous-image qui contient le template que nous voulons trouver. Cette technique est répétée pour l'ensemble de l'image, et le point qui mène à une meilleure correspondance est défini comme étant le point où la forme (donnée par le modèle) se trouve à l'intérieur de l'image. Les modèles sont généralement utilisés pour imprimer des caractères, identifier des nombres et d'autres petits objets simples. Il peut être utilisé pour la détection et la reconnaissance des panneaux routiers. Template Matching est une stratégie pour découvrir des zones d'une image qui correspond à une image de modèle (patch). La technique d'appariement ne prend pas seulement la mesure de similarité mais calcule l'imprécision entre les images en prenant en compte la différence au moyen de la métrique MSE (Mean Squared Error) [157].

Les techniques de correspondance de modèles sont flexibles et relativement simples à utiliser, ce qui en fait l'une des méthodes les plus populaires de localisation d'objets. Leur applicabilité est limitée principalement par la complexité de calcul, car l'identification des modèles volumineux et complexes peut prendre beaucoup de temps. Nous avons testé template matching pour la reconnaissance des panneaux routiers et nous avons remarqué que la méthode n'est pas robuste aux changements d'échelle et aux rotations, ainsi qu'elle requiert une puissance de calcul élevée, car la détection de grands motifs d'une image prend beaucoup de temps. C'est ce qui rend cette méthode inutile pour la reconnaissance des panneaux routiers.

## 5.5 SVM (Support Vector Machine)

Les machines à vecteurs de support (SVM) ont été présentés par Cortes et Vapnik [158], c'est une méthode d'apprentissage relativement nouvelle utilisée pour la classification binaire. L'idée de base est de trouver un hyperplan qui sépare parfaitement les données dimensionnelles en deux classes. Cependant, étant donné que souvent les données ne sont pas linéairement séparables, les SVM introduisent la notion d'un «espace de caractéristiques induit par le noyau» qui transpose les données dans un espace de dimension supérieure où les données sont séparables. En règle générale, projeter les données dans un tel espace causerait des problèmes de calcul et surapprentissage. L'idée clé utilisée dans les SVM est que l'espace de plus grande dimension n'a pas besoin d'être traité directement (il s'avère que seule la formule pour le produit scalaire dans cet espace est nécessaire), ce qui élimine les préoccupations ci-dessus. De plus, la dimension VC<sup>1</sup> des SVM peut être explicitement calculée, contrairement à d'autres méthodes d'apprentissage comme les réseaux de neurones, pour lesquels il n'existe aucune mesure. Dans l'ensemble, les SVM sont intuitifs, théoriquement bien fondés et se sont révélés pratiquement efficaces [159].

Étant donné  $l$  exemples d'apprentissage  $x_i, y_i, i = 1, \dots, l$  où chaque exemple a  $d$  entrées  $x_i \in R^d$ , et une étiquette de classe avec l'une des deux valeurs ( $x_i \in R^d$ ). Maintenant, tous les hyperplans dans  $R^d$  sont paramétrés par un vecteur ( $w$ ), et une constante ( $b$ ), exprimée dans l'équation :

$$W.X + b = 0 \tag{5.10}$$

(Rappelons que  $w$  est en fait le vecteur orthogonal à l'hyperplan.) Étant donné un tel hyperplan ( $w, b$ ) qui sépare les données et qui classe correctement les données d'apprentissage, ceci donne la fonction suivante :

$$f(x) = \text{sign}(w.x + b) \tag{5.11}$$

Cependant, un hyperplan donné représenté par  $w$  et  $b$  est également exprimé par toutes les paires ( $w, b$ ). Donc l'hyperplan canonique est celui qui sépare les données de l'hyperplan par une «distance» d'au moins 1 :

$$\begin{cases} x_i.w + b \geq +1 & \text{si } y_i = +1 \\ x_i.w + b \leq -1 & \text{si } y_i = -1 \end{cases} \tag{5.12}$$

---

1. une mesure de la probabilité qu'un système fonctionne bien sur des données invisibles



détails.) Le problème est finalement transformé en :

$$\begin{aligned}
 \text{Minimizer :} \quad W(\alpha) &= - \sum_{i=1}^l \alpha_i + \frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l y_i y_j \alpha_i \alpha_j (x_i, x_j) \\
 \text{Soumis à :} \quad & \sum_{i=1}^l y_i \alpha_i = 0 \\
 & 0 \leq \alpha_i \leq C \quad (\forall i)
 \end{aligned} \tag{5.15}$$

où  $\alpha$  est le vecteur de  $l$  multiplicateurs de Lagrange non négatifs à déterminer, et  $C$  est une constante. Donc la matrice  $(H)_{ij} = y_i y_j (x_i, x_j)$ .

Ce problème de minimisation est appelé problème de programmation quadratique (QP). Heureusement, de nombreuses techniques ont été développées pour les résoudre. De plus, à partir de la dérivation de ces équations, il a été vu que l'hyperplan optimal peut être écrit comme suit [160] :

$$W = \sum_{i=1}^l \alpha_i y_i x_i \tag{5.16}$$

C'est-à-dire que le vecteur  $W$  est juste une combinaison linéaire des exemples d'apprentissage comme suit :

$$\alpha_i (y_i (w \cdot x_i + b) - 1) = 0 \quad \forall i \tag{5.17}$$

Ce qui est juste une façon de dire que lorsque la distance fonctionnelle d'un exemple est strictement supérieure à 1 (quand  $y_i (w \cdot x_i + b) > 1$ ), alors  $\alpha_i = 0$ . Ainsi, seuls les points de données les plus proches contribuent à  $w$ . Ces exemples d'apprentissage pour lesquels  $\alpha_i > 0$  sont appelés **vecteurs de support**. Ils sont les seuls nécessaires pour définir (et trouver) l'hyperplan optimal. Intuitivement, les vecteurs de support sont les «cas limites» dans la fonction de décision. Il est encore plus intéressant de noter que  $\alpha_i$  peut être considéré comme un «indice de difficulté» pour l'exemple  $x_i$  à quel point cet exemple était important dans la détermination de l'hyperplan [159].

Donc, pour la contrainte  $0 \leq \alpha_i \leq C \quad (\forall i)$ , lorsque  $C = \infty$ , l'hyperplan optimal sera le seul qui sépare complètement les données (en supposant qu'il en existe un). Pour le  $C$  fini, ceci change le problème en trouvant un classificateur "à marge douce", qui permet de classifier incorrectement certaines données.  $C$  est peut être considéré comme un paramètre accordable : une valeur de  $C$  élevée correspond à plus d'importance pour

classer correctement toutes les données d'entraînement, une valeur de  $C$  inférieure produit un hyperplan «plus flexible» qui tente de minimiser l'erreur de marge (à quel point  $y_i(w \cdot x_i + b) < 1$ ) pour chaque exemple. Les valeurs finies de  $C$  sont utiles dans les situations où les données ne sont pas linéairement séparables (peut-être parce que les données d'entrée  $X_i$  sont bruitées) [162].

Si un ensemble de données n'est pas linéairement séparable, cela ne signifie pas qu'il n'y a pas d'autre moyen pour séparer les données. Par exemple, il peut être plus facile de séparer les données en utilisant des courbes polynomiales ou des cercles. Cependant, trouver la courbe optimale pour s'adapter aux données est difficile, et il serait dommage de ne pas utiliser la méthode de recherche de l'hyperplan optimal. En effet, il existe un moyen de "pré-traiter" les données de telle sorte que le problème se transforme en un problème de recherche simple. Pour ce faire, soit un mappage  $z = \phi(x)$  qui transforme le vecteur d'entrée  $x$  de dimension  $D$  en un vecteur  $z$  de dimension  $D'$ . Il faut choisir un  $\phi()$  pour que les nouvelles données d'entraînement  $\phi(x_i), y_i$  soient séparables par un hyperplan [163].

Mais le problème est comment choisir  $\phi()$ ? Il faudrait beaucoup de travail pour en construire un explicitement pour tout ensemble de données. Si  $\phi(x)$  projette le vecteur d'entrée dans un espace suffisamment grand ( $D' \gg D$ ), les données devraient éventuellement être séparables. Alors peut-être il y a un standard  $\phi()$  qui fait cela pour la plupart des données. Mais couler dans un très grand espace dimensionnel est aussi inquiétant. Rappelons que la construction de la matrice  $H$  nécessite les produits scalaires  $(x_i \cdot x_j)$ . Si  $D'$  est exponentiellement plus grand que  $D$  (et cela pourrait très bien être le cas), le calcul de  $H$  devient prohibitif (sans parler des besoins d'espace supplémentaire). En outre, en augmentant la complexité du système de telle manière, le surapprentissage devient une préoccupation. En projetant les données dans un espace dimensionnel suffisamment élevé, il est possible de séparer n'importe quel ensemble de données [160]. Comment pouvons-nous être sûrs que le système ne correspond pas seulement aux particularités des données d'apprentissage, mais apprend en fait un modèle légitime qui généralisera à d'autres données sur lesquelles il n'a pas été entraîné?

Les SVM évitent ces problèmes, étant donné une application  $z = \phi(x)$ , pour mettre en place notre nouveau problème d'optimisation, nous remplaçons simplement toutes les occurrences de  $x$  par  $\phi(x)$ . Donc l'équation 5.16 serait :

$$W = \sum_{i=1}^l \alpha_i y_i \phi(x_i) \tag{5.18}$$

Et l'équation 5.11 serait :

$$\begin{aligned} f(x) &= \text{sign}(w \cdot \phi(x) + b) \\ &= \text{sign}\left(\left[\sum_{i=1}^l \alpha_i y_i \phi(x_i)\right] \cdot \phi(x) + b\right) \\ &= \text{sign}\left(\sum_{i=1}^l \alpha_i y_i (\phi(x_i) \cdot \phi(x)) + b\right) \end{aligned} \quad (5.19)$$

L'observation importante dans tout cela, c'est que chaque fois qu'un  $\phi(x_i)$  apparaît, il est toujours dans un produit scalaire avec un autre  $\phi(x_j)$ . Autrement dit, si nous connaissons la formule du noyau (Kernel) pour le produit scalaire dans l'espace des caractéristiques dimensionnelles supérieures :

$$K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j) \quad (5.20)$$

nous n'aurions jamais besoin de traiter directement le mapping  $z = \phi(x)$ . La matrice d'optimisation serait simplement  $(H)_{ij} = y_i y_j (K(x_i, x_j))$  et le classifieur  $f(x) = \text{sign}(\sum_i \alpha_i y_i (K(x_i, x)) + b)$ . Une fois que le problème est mis en place de cette manière, trouver l'hyperplan optimal se poursuit comme d'habitude, seul l'hyperplan sera dans un espace caractéristique inconnu. Dans l'espace d'entrée d'origine, les données seront séparées par un contour courbe, éventuellement non continu.

La solution du problème dépend uniquement du noyau  $K(x_i, x_j)$ , des noyaux utiles ont déjà été découverts comme le noyau linéaire, polynomiale, radiale (RB) ou Sigmoïde.

Nous choisissons le SVM avec un noyau linéaire comme un classifieur pour la reconnaissance des panneaux routiers dû à sa rapidité en temps d'exécution, performances, efficacité et simplicité. Nous avons utilisé deux descripteur de caractéristiques LBP (Local Binary pattern) et HOG (Histogram of Oriented Gradient) avec le SVM, nous avons remarqué que le descripteur HOG est plus performant que LBP à décrire les caractéristiques discriminantes des panneaux routiers.

## 5.6 L'histogramme des gradients orientés

Le descripteur HOG proposé par Dalal et al [89] est très efficaces pour la tâche de description et de détection d'objet, surtout en combinaison avec un classifieur SVM linéaire. HOG «Histograms of Oriented Gradients» comme son nom l'indique, le descripteur est constitué d'histogrammes d'orientation des gradients d'images, extraits de blocs d'image représentant les objets d'intérêt. En revanche, les caractéristiques HOG

sont généralement extraites d'une grille de «cellules», dans le cadre d'une fenêtre glissante. Par conséquent, ils décrivent un objet dans sa totalité et sa forme et ne se limitent pas à son apparence. Cela les rend très appropriés et efficaces pour la classification par un SVM [164].

Les histogrammes des gradients orientés sont des caractéristiques de haut niveau qui utilisent le gradient de l'image pour décrire un bloc d'image. Le descripteur construit est ensuite utilisé pour former le classifieur. Les étapes d'extraction des caractéristiques HOG sont résumées comme suit :

- Calculer le gradient de l'image : il est important de noter que appliquer un filtre gaussien ou similaire à l'image avant de calculer le gradient est déconseillé par Dalal et Triggs [89]. Ils ont également constaté que tout autre noyau utilisé pour calculer les gradients, comme le filtre Sobel, réduisait également les performances. Généralement, les gradients sont calculés à partir de l'intensité d'une image, mais pour une détection plus discriminante, il est possible d'utiliser les trois canaux RGB. Le gradient est calculé à l'aide de l'équation suivante [89] :

$$\nabla I(x, y) = \begin{pmatrix} I_x(x, y) \\ I_y(x, y) \end{pmatrix} \approx \begin{pmatrix} I(x+1, y) - I(x-1, y) \\ I(x, y+1) - I(x, y-1) \end{pmatrix} \quad (5.21)$$

$I(x, y)$  est l'intensité de l'image  $I$  aux coordonnées  $(x, y)$ , Le gradient est ensuite converti en forme polaire avec les équations suivantes :

$$\|\nabla I(x, y)\|_2 = \sqrt{I_x^2 + I_y^2} \quad (5.22)$$

$$\theta_{\nabla I(x, y)} = \text{atan2}(I_y, I_x) \in (-\pi, \pi) \quad (5.23)$$

$$\text{atan2}(x, y) = \begin{cases} \arctan \frac{y}{x} & x > 0 \\ \arctan \frac{y}{x} + \pi & x < 0 \text{ et } y \geq 0 \\ \arctan \frac{y}{x} - \pi & x < 0 \text{ et } y < 0 \\ +\frac{\pi}{2} & x = 0 \text{ et } y > 0 \\ -\frac{\pi}{2} & x = 0 \text{ et } y < 0 \\ \text{Non définie} & x = 0 \text{ et } y = 0 \end{cases} \quad (5.24)$$

- La formation des cellules : Une fois le gradient est calculé et transformé en forme polaire, les images de magnitude et d'orientation sont scanées par une fenêtre glissante. La fenêtre est ensuite divisée en cellules d'une certaine taille. Dalal et Triggs ont constaté que l'algorithme fonctionnait mieux avec des tailles de cel-

lules de  $6 \times 6$  pixels et  $8 \times 8$  pixels. La hiérarchie des divisions (y compris les blocs) est illustrée dans la figure 5.9

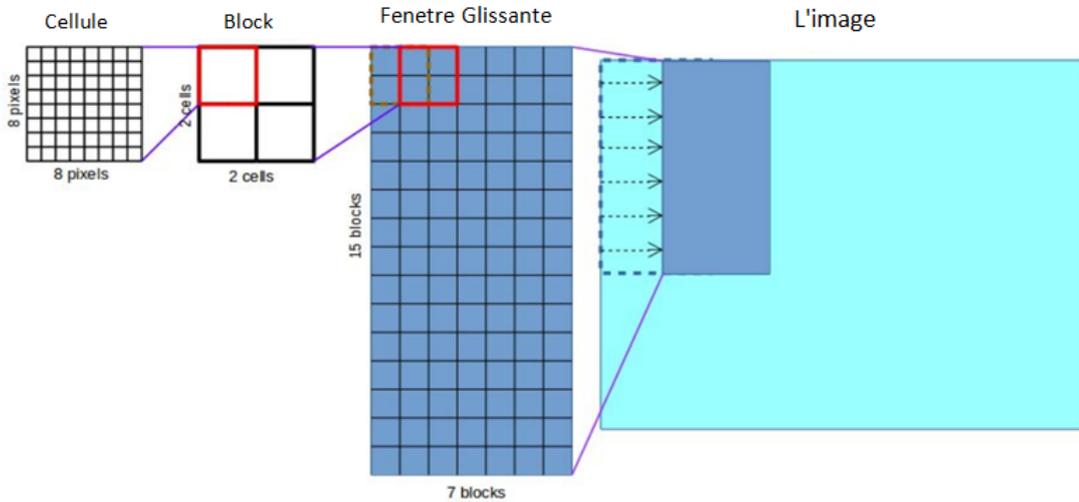


FIGURE 5.9 – La hiérarchie des divisions pour calculer HOG [11].

Ensuite, des histogrammes d'orientation de gradient peuvent être calculés pour chaque cellule. Le choix de la longueur de l'histogramme est important et dépend de la plage d'orientation. L'histogramme dans sa forme la plus simple est défini par l'équation suivante :

$$H(i) = \sum_{x',y'=0}^{x,y} M(x', y', i) \quad 0 \leq i < Li, L \in \mathbb{N} \quad (5.25)$$

Où

$$M(x', y', i) = \begin{cases} \|\nabla I(x', y')\|_2 & h(\text{atan2}(I_{y'}, I_{x'})) = i \\ 0 & \text{Autres} \end{cases} \quad (5.26)$$

h est une fonction de quantification, généralement représentée par :

$$h(x) = \lfloor \frac{x}{2\pi} + \frac{1}{2} \rfloor \times L \quad (5.27)$$

- Formation de bloc et normalisation d'histogramme : une fois les histogrammes calculés, les cellules sont regroupées en blocs qui se chevauchent, de sorte que chaque bloc partage la majorité avec ses voisins, mais exclut au moins une ligne ou une colonne de cellules. L'ordre des cellules n'a pas d'importance, tant qu'il est identique pendant l'entraînement et la détection. Ce processus introduit non

seulement de la redondance, mais permet également de normaliser les histogrammes autour de toute une région. Dalal et Triggs ont constaté que les tailles des blocs de  $2 \times 2$  et  $3 \times 3$  cellules étaient les plus optimales en ce qui concerne la taille des cellules [11].

Les histogrammes des cellules correspondantes sont concaténés pour former un vecteur descripteur. Ce descripteur est ensuite normalisé en utilisant l'une des relations suivantes :

$$L2 - norm : x = \frac{v}{\sqrt{\|v\|_2^2 + e^2}} \quad (5.28)$$

$$L1 - norm : x = \frac{v}{\|v\|_1 + e} \quad (5.29)$$

$$L1 - sqrt : x = \sqrt{\frac{v}{\sqrt{\|v\|_2^2 + e^2}}} \quad (5.30)$$

Où  $v$  est le vecteur d'histogramme de bloc,  $\|v\|_k$  est son  $k$  vecteur normalisé pour  $k=1,2$ . La constante  $e$  doit être petite, mais sa valeur exacte n'est pas spécifiée.

## 5.7 Conclusion

Dans ce chapitre, nous avons présenté brièvement les techniques de reconnaissance des panneaux routiers. Nous avons testé les points d'intérêt SIFT et SURF, le template matching et le SVM. L'inconvénient des points d'intérêt est qu'elles ne sont pas robustes aux changements de luminosité ainsi qu'elle requiert une puissance de calcul élevée. La méthode de template matching n'est pas robuste aux changements d'échelle et aux rotations. Nous avons choisi le SVM linéaire car il est plus rapide et efficace, nous avons testé deux caractéristiques LBP et HOG avec le SVM et nous avons constaté que le SVM linéaire avec le descripteur HOG donne les meilleurs résultats.

## **Chapitre 6**

# **Systeme de détection et de reconnaissance des panneaux de limitation de vitesse**

### 6.1 Introduction

Les panneaux de signalisation fournissent des informations utiles pour réguler le trafic, cependant, les conducteurs peuvent parfois ne pas tenir compte d'un panneau particulier en raison de la fatigue au volant, de la conduite dans des conditions météorologiques menaçantes ou de la recherche d'une adresse. Parfois, les conducteurs conduisent plus vite que la limite de vitesse, par conséquent, non seulement la vitesse augmente le risque d'accidents, mais elle augmente également le risque de blessures graves et de décès en cas d'accident. Comme l'indique l'ONISR (Observatoire National Interministériel de la Sécurité Urbaine), la vitesse (inapproprié ou excessif) peut être considéré comme l'une des causes principales des accidents mortels. La capacité de perception visuelle diminue au fur et à mesure que la vitesse augmente, ainsi que la distance de freinage en cas de danger. Un piéton dispose d'un champ visuel de 180°, un automobiliste roulant à 40 km/h de 100°, à 130 km/h, l'angle d'ouverture n'est plus que de 30°. Une vision et une marge de réaction limitées rendent le traitement et l'analyse des informations plus complexes et peuvent être à l'origine de situations dangereuses. De plus, la gravité de l'accident dépend de l'énergie du choc et donc de la vitesse des véhicules. Outre l'aspect humain, le coût économique représenté est considérable tant pour la société que pour les conducteurs. L'intérêt de proposer un système de reconnaissance automatique des panneaux de limitation de vitesse (PLV) est alors primordiale. Le concept est déjà à l'œuvre dans de nombreux véhicules récents grâce à l'utilisation du régulateur de vitesse. Ces systèmes soulagent le conducteur qui peut ainsi reporter son attention sur la route plutôt que sur son compteur mais doivent être activés et mis à jour manuellement. Entièrement automatiques, ils amélioreraient notablement le confort des utilisateurs [165].

Le nombre croissant des projets liés à cette thématique témoigne bien de l'intérêt grandissant pour ce domaine, tant dans la recherche académique que dans le monde industriel. Depuis quelques années, des véhicules sont même équipés de modules de détermination de limitation de vitesse de plus en plus sophistiqués. La majorité reposent sur une caméra monoculaire qui détecte et reconnaît les panneaux présents sur les bords des routes. Ces derniers correspondent à des panneaux de limitation de vitesse locales ou temporaires, dénotant la présence de zones dangereuses. Dans ce chapitre nous présentons notre système de reconnaissance des panneaux de limitation de vitesse.

### 6.2 État de l'art

Pour accroître la sécurité de la conduite, un système de reconnaissance des panneaux de limitation de vitesse devient indispensable pour réduire le nombre de victimes de la route. La reconnaissance des panneaux de limitation de vitesse est l'un des éléments importants pour réaliser un ADAS (Advanced Driver Assistance System)

## Chapitre 6 : Système de détection et de reconnaissance des panneaux de limitation de vitesse

---

amélioré. Généralement, le système de reconnaissance des panneaux de limitation de vitesse comporte deux parties : la détection des panneaux dans l'image puis les valeurs de limitation de vitesse sont reconnus dans l'étape de classification.

Plusieurs travaux ont été portés sur la reconnaissance des panneaux de vitesse. Les auteurs de [166] ont proposé un framework en temps réel pour détecter et reconnaître les PLV dans les réseaux ferroviaires, deux SVM différents sont formés, le premier est pour reconnaître les types des signes et le second pour reconnaître les chiffres, le framework a réalisé un rappel d'environ 95% sur les vidéos de chemin de fer. Une version modifiée de l'histogramme des gradients orientés (HOG) est utilisée avec SVM par [167] pour détecter et reconnaître plus de 94% des panneaux de limite de vitesse nord-américains. Les auteurs de [168] ont utilisé des informations de forme et d'intensité pour détecter les PLV américaines après avoir identifié les régions candidates comme des régions extrêmes extrêmement stables (MSER), ils ont utilisé le filtre de Kalman pour suivre les panneaux détectés, cependant, seuls les mouvements linéaires de la voiture sont considérés dans la phase de suivi. Au stade de la classification, un réseau de neurone artificiel est utilisé pour la reconnaissance avec une précision de 98% sur 12300 images. Les fonctions HOG et MSER sont également utilisées par [169] pour la détection et la classification des PLV, un taux de classification de 93,67% est obtenu avec un temps de traitement de 10,75 ms. Un nouveau descripteur de forme CtC (centroid-to-contour) est proposé par [170] pour détecter les PLV, puis un SVM est utilisé pour classer les panneaux détectés en temps réel qui peut aller jusqu'à 22 fps.

Gomes [171] a proposé une cascade de classifieurs boostés et entraînés sur les caractéristiques de haar-like pour détecter les PLV. Dans la phase de classification, ils ont comparé sept techniques d'apprentissage automatique : le OPF (Optimum Path Forest), SVM, perceptron multicouche, k-plus proche voisin (kNN), machine d'apprentissage extrême, moindres carrés et techniques d'apprentissage machine des moindres carrés) et ils ont prouvé que le classifieur SVM avec noyau linéaire avait la précision moyenne la plus élevée de 99,82% et l'écart-type le plus faible, Cependant, OPF avec la distance euclidienne a eu le temps de test le plus bas, son temps de test était 64 fois plus bas que le SVM, mais aussi avec des performances moins que le SVM.

Liu, B. et al [172] combinent la cartographie log-polaire et le LLC (Locality-constrained Linear Coding) pour reconnaître les PLV, ils atteignent une précision de 97,31% sur l'ensemble de données GTSRB. Une méthode d'illumination robuste pour la détection PLV en temps réel est proposée par [173] en utilisant MCT (Modified Census Transform) et SVM, ils ont atteint un taux de détection de 99,8% et un taux de reconnaissance de 98,4% .

Dans le but de développer un système robuste à la variabilité des couleurs et aux variations d'éclairage, Moutarde et al [174] ignoraient les informations de couleur, ils se basaient uniquement sur la forme, ce qui augmente le nombre de fausses alarmes. Ils ont atteint un taux de détection d'environ 90% pour les panneaux de limitation de vitesse américains et européens avec un temps de traitement 20 images/seconde.

## Chapitre 6 : Système de détection et de reconnaissance des panneaux de limitation de vitesse

---

Un réseau de neurones est utilisé par Ishak, K. A. et al [175], Eichner, M.L. et al [176] et Miyata, S. [177] pour la classification des signes de limitation de vitesse. CNN (Convolutional Neural Network) est utilisé par Yan, G., et al [178] dans l'étape de classification et ils ont atteint un taux de reconnaissance d'environ 98,51 %. Peemen, M., et al [179] utilisent aussi CNN pour la détection et la classification des panneaux de limitation de vitesse, cependant, ils ont obtenu un grand nombre de fausses alarmes lors du premier test du CNN formé. Pour augmenter la précision du système, ils ont utilisé un algorithme de boosting itératif pour apprendre de l'erreur précédente mais il a consommé plus de temps de traitement ce qui rend le système inadapté aux appareils simples. CNN est également utilisé par [180] en appliquant le pré-traitement et le post-traitement sur les images pour améliorer les performances, ils ont atteint une précision supérieure à 97 % sur l'ensemble de données LISA-TS.

Comme nous avons illustré dans Saadna, Y. et al [58] le problème principal dans les système de reconnaissance des panneaux de signalisation ce n'est pas comment atteindre le plus haut rappel, cependant, obtenant également une haute précision et développer un système scalable qui prouve les mêmes performances sur différentes bases de données. Dans ce chapitre nous proposons un système robuste, efficace et scalable pour la détection et la reconnaissance des panneaux de limitation de vitesse avec non seulement un rappel élevé, mais aussi une grande précision.

### 6.3 Les bases de données disponibles

Il existe quelques bases de données des panneaux routiers comme :

- GTSRB (German Traffic Sign Benchmark) est créée pour la compétition allemande sur la reconnaissance des panneaux routiers en 2013 (the German traffic sign recognition benchmark). Elle est largement utilisée dans le domaine de la reconnaissance des panneaux routiers. Ce qui facilite aux auteurs de comparer leurs approches proposées avec l'état de l'art. Elle comprend deux ensembles de données : GTSD (German Traffic Sign Detection Benchmark) [92], qui contient 900 images de taille 1 360 x 800 pixels, et GTSRB (German Traffic Sign Recognition Benchmark) [97] qui contient 39 209 images d'apprentissage et 12 630 images de test, la taille des images varie entre 15x15 et 250x250 pixels pour évaluer la phase de classification.
- BTSD (Belgium Traffic Sign Dataset) [181] : elle contient 25 630 images de taille 1 628x1 236 pixels, il y a aussi la base BTSC (Belgium Traffic Sign Classification) est une extraction des régions d'intérêt contenant des panneaux de signalisation routière dans la base BTSD, elle est composée de 4 591 images d'apprentissage et 2 534 images de test.
- STS (Swedish Traffic Sign) [83] : elle contient plus de 20 000 images de taille 1 280x960. Les panneaux de limitation de vitesse dans cette base sont entourés d'un cercle rouge, comme dans les GTSB et BTSD. Toutefois, le fond du panneau est jaune

## Chapitre 6 : Système de détection et de reconnaissance des panneaux de limitation de vitesse

en STS et blanc en GTSB et BTSD. Comme nous utilisons le descripteur HOG qui traite les images au niveaux de gris, les panneaux apparaissent de la même manière dans les images au niveaux de gris.

- Stereopolis [182] est composé de 847 images de taille de 1 920x1 080 pixels.
- LISA dataset ( Laboratory for Intelligent and Safe Automobiles) [183] est composé de 7 855 images de taille 1 024x522. Elle contient également des vidéos et des images annotées capturées aux États-Unis.
- La base de données RUG[184] : ne contient que trois classes des panneaux des Pays-Bas, elle est composé de 48 images de taille 360x270 pixels.

Les bases de données disponibles sont résumées dans le tableau 6.1 :

TABLE 6.1 – Les bases de données des panneaux routiers.

Base	Nombre des images	Taille des image	Pays
GTSRB	51 839	15x15 à 250x250	Allemagne
GTSDDB	900	1 360x800	Allemagne
BTSD	25 634	1 628x1 236	Belgique
STS	20 000	1 280x960	Suède
RUG	48	360x270	Pays-Bas
Stereopolis	847	1 920x1 080	France
LISA	7 855	1 024x522	États Unis

Nous choisissons les bases GTSB, BTSD et STS pour tester notre système proposé en raison de la diversité de leurs classes et de leur contenu, de leurs grandes tailles et de leur utilisation répandue par la plupart des approches, comme nous l’avons vu dans [185] et [58]. Le GTSB était soumis à la concurrence allemande de référence en matière de reconnaissance des panneaux routiers, ce qui facilitait l’analyse comparative de diverses méthodes. Il existe d’autres grandes base de données comme la base LISA, mais elle n’adhère pas à la convention de Vienne et n’a donc pas les mêmes caractéristiques que GTSB et BTSD. Les panneaux de limitation de vitesse dans la base LISA sont gris et carrés, tandis que dans GTSB, ils sont des cercle rouge.

### 6.4 Architecture du système proposé

L’architecture de notre système de détection et de reconnaissance des panneaux de limitation de vitesse proposée comprend quatre étapes (voir figure 6.1) : Localisa-

## Chapitre 6 : Système de détection et de reconnaissance des panneaux de limitation de vitesse

tion des régions d'intérêt rouges, détection des formes circulaire, reconnaissance des panneaux de limitation de vitesse et classification des chiffres. Dans l'étape de localisation, une nouvelle méthode de segmentation est proposée, la transformée Circle Hough (CHT) est utilisée dans l'étape de détection pour détecter les cercles. Une fois les régions d'intérêt détectées, un SVM linéaire entraîné seulement sur la base de données GTSRB est utilisé pour reconnaître les panneaux de limitation de vitesse parmi les régions détectées. Enfin, un SVM linéaire entraîné sur la base de données MNIST est utilisé pour reconnaître les chiffres des panneaux de limitation de vitesse.

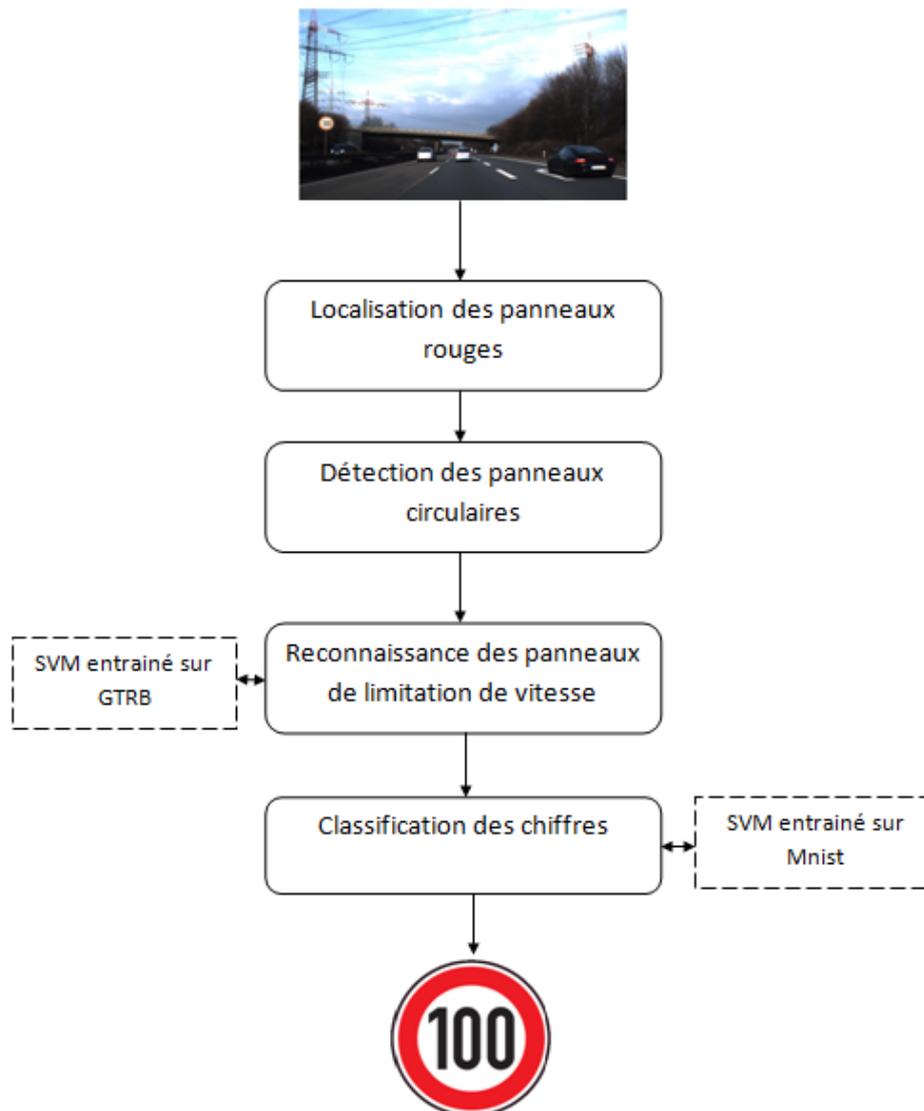


FIGURE 6.1 – Architecture du système de reconnaissance des panneaux de limitation de vitesse.

## 6.5 Approche proposée

### 6.5.1 Localisation des panneaux de limitation de vitesse

Nous avons décrit dans le chapitre 3 que les méthodes de détection sont classées en trois catégories, les méthodes basées sur la couleur, les méthodes basées sur la forme et les méthodes basées sur l'apprentissage. Ces dernières donnent de bons résultats, cependant, elles nécessitent de grandes bases de données d'entraînement. Ainsi, leur principale faiblesse est que les performances du système sont fortement liées à la base d'apprentissage et ne donneront pas les mêmes performances avec d'autres bases de test. Les méthodes basées sur la forme sont coûteuses en temps de calcul. Cependant, les méthodes de segmentation sont sensibles aux changements d'illumination et d'autres facteurs, mais elles sont simples et rapides, c'est pourquoi nous choisissons la couleur comme caractéristique discriminante pour localiser les panneaux de limitation de vitesse en temps réel. Nous avons choisi l'espace RGB car la transformation de RGB en HSV est non linéaire, Gmez-Moreno, H. et al. [77] ont prouvé que la standardisation de l'espace colorimétrique RVB donne de bonnes performances avec moins d'opérations que le HSV. Alors, pourquoi utiliserions-nous une transformation non linéaire si une simple normalisation est suffisante ?

Dans l'étape de localisation, comme la couleur des panneaux de limitation de vitesse est rouge, nous proposons une nouvelle méthode de segmentation pour renforcer l'intensité des pixels rouges et éliminer autant que possible les pixels indésirables. Comme est illustré dans le chapitre 3 section 3.3, l'idée est de multiplier le résultat de l'équation 3.18 par la valeur minimale entre la différence de canal rouge et canal bleu, et la différence de canal rouge et canal vert  $\min(r - g, r - b)$  (les valeurs négatives sont mises à zéro et celles supérieures à 255 sont mises à 255). Avec cette équation plus la couleur est un vrai rouge, plus la différence minimale est grande ce qui augmente l'intensité de ces pixels. Et plus la couleur n'est pas un vrai rouge plus la différence minimale est petite ce qui diminue l'intensité.

$$R = r - \frac{(g + b)}{2} \times \min(r - g, r - b) \quad (6.1)$$

### 6.5.2 Détection des panneaux de limitation de vitesse

Dans l'étape de détection nous proposons une variante de LBP et l'utiliser comme un filtre et non pas comme un extracteur de caractéristiques. La méthode proposée est utilisée pour détecter les régions homogènes afin de réduire le nombre des faux positifs. La contribution comme décrit dans le chapitre 4 section 4.3 est de donner le même poids  $n$  pour toutes les différences, comme il y a 8 voisins et pour obtenir la valeur maximale (255) dans le pixel  $(x_c, y_c)$  on met  $n = 5$  donc  $(8 * (2^5) = 256)$  tel qu'il

## Chapitre 6 : Système de détection et de reconnaissance des panneaux de limitation de vitesse

---

est illustré dans l'équation suivante :

$$LBP_{P,R}(x_c, y_c) = \sum_{p=0}^{P-1} 2^5 s(g_p - g_c) - 1 \quad (6.2)$$

Pour calculer des régions homogènes nous considérons seulement les pixels similaires avec des intensités approximatives selon un seuil  $th$  et c'est pourquoi nous calculons la valeur absolue de  $(g_p - g_c)$ , nous définissons la fonction  $s(x)$  comme suit :

$$s(x) = \begin{cases} 1 & \text{si } |x| \geq th \\ 0 & \text{sinon} \end{cases} \quad (6.3)$$

Comme les panneaux de limitation de vitesse sont circulaire, nous utilisons la transformé de hough circulaire pour détecter les cercles dans les images segmentées vue sa rapidité et sa robustesse aux bruits.

### 6.5.3 Classification des panneaux de limitation de vitesse

Une fois que les cercles rouges sont détectés, maintenant il est temps de les classifier. Gomes, SL, et al [171] ont prouvé que le classifieur SVM avec un noyau linéaire a la plus haute précision et l'écart-type le plus faible dans la classification des panneaux de limitation de vitesse comparé au SVM polynomial, KNN (K-Nearest Neighbor), OPF (Optimum-Path Forest), MLP (Multi - Layer Perceptron), ELM (Extreme Learning Machine), LMS (Least Mean Squares) et LS (Least Squares), c'est ce qui nous a fait choisir le SVM linéaire comme classificateur. Nous utilisons un premier SVM qui détermine si les cercles détectés sont des panneaux de limitation de vitesse ou non. La base d'apprentissage du premier SVM est extrait de la base GTSRB, elle comprend deux classes : la classe positive qui contient 900 images des panneaux de limitation de vitesse, la classe négative comprend 700 images des autres panneaux d'interdiction, des fausses alarmes possibles comme les lumières de voiture, les feux de circulation et les panneaux publicitaires, la figure 6.2 et 6.3 illustre des exemples d'images de la base d'apprentissage du premier SVM.

À ce stade, les panneaux détectés sont filtrés et les faux positifs sont éliminés, maintenant, on doit reconnaître les chiffres des panneaux de limitation de vitesse. Nous utilisons les histogrammes verticale et horizontale de l'image ROI détectée pour localiser les chiffres des panneaux et si le premier chiffre est différent de 1 donc le deuxième est automatiquement un 0. Pour identifier les chiffres des panneaux de limitation de vitesse nous utilisons un deuxième SVM entraîné sur la base MNIST, nous choisissons

## Chapitre 6 : Système de détection et de reconnaissance des panneaux de limitation de vitesse



FIGURE 6.2 – Exemple des images de la classe positive du premier SVM.



FIGURE 6.3 – Exemple des images de la classe négative du premier SVM.

la base MNIST pour ces raisons :

- Dans la base GTSRB il n'y a pas assez de panneaux de limitation de vitesse pour former un classifieur robuste, il y a des types de panneaux qui contiennent qu'une vingtaine d'images comme les panneaux de vitesse 20.
- Absence des panneaux de limitation de vitesse 40, 90 dans la base GTSRB, car la limitation de vitesse sur les routes non urbaine en Allemagne est 100 km/h, par contre dans d'autres pays comme France, Belgique, Italy et autres pays la vitesse limitée dans les routes non urbaine est 90.
- Les chiffres des panneaux de limitations de vitesse ne sont pas toujours clairs et lisibles, à cause de vitesse, manque de lumière ou vibration comme est illustré dans la figure 6.4
- Absence des panneaux de limitation de vitesse 130 dans la base GTSRB, parce que dans Allemagne il n'y a pas de limitation de vitesse dans les autoroutes, cependant dans les autres pays la vitesse limitée dans les autoroutes est 130 km/h comme est illustré dans la figure 6.5.

La base d'apprentissage du deuxième SVM contient 9 classes (les chiffres de 1 à 9), chaque classe comprend 100 images extraites de la base MNIST. Nous n'utilisons pas toute la base MNIST mais juste les chiffres similaires aux chiffres des panneaux routiers comme est montré dans la figure 6.6.

## Chapitre 6 : Système de détection et de reconnaissance des panneaux de limitation de vitesse



FIGURE 6.4 – Exemple des panneaux flous.

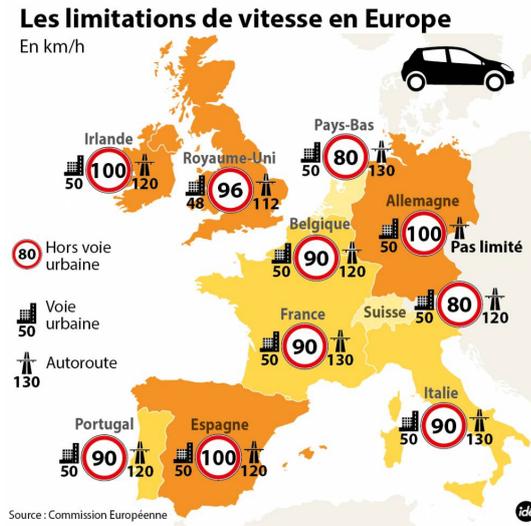


FIGURE 6.5 – Limitation de vitesse en europe.

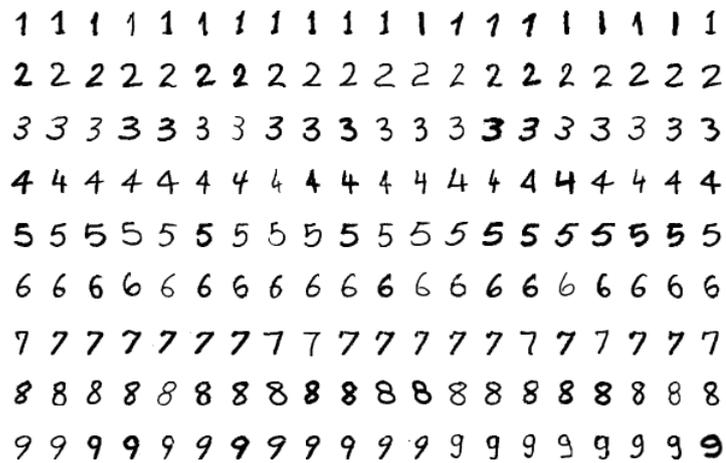


FIGURE 6.6 – Exemple de la sous base MNIST utilisée.

## Chapitre 6 : Système de détection et de reconnaissance des panneaux de limitation de vitesse

---

L'algorithme suivant décrit notre système proposé :

---

### Algorithm 2 Reconnaissance des panneaux de limitation de vitesse

---

**Require:** *img* : image de taille ( $H, W$ );  $r, g, b$  : les intensités rouge, vert et bleu du pixel  $img(i, j)$ .

```
for  $i=0$  to  $H$  do
  for  $j=0$  to  $W$  do
     $R(i, j) = r - \frac{(g+b)}{2} \times \min(r - g, r - b)$ ;
  end for
end for
for  $i=0$  to  $H$  do
  for  $j=0$  to  $W$  do
     $I(i, j) = \sum_{i'=i-k, j'=j-k}^{i+k, j+k} 2^5 s(R(i', j') - R(i, j)) - 1$ ;
  end for
end for
vecteur_ROI = Hogh_cercles_detection(I);
for  $i=0$  to sizeof(vecteur_ROI) do
  if (SVM1.predict(vecteur_chiffre(0))) then
    Calculer l'histogramme horizontal et vertical;
    Localiser les chiffres;
    vecteur_chiffre = detecter_chiffre();
    if (SVM2.predict(vecteur_chiffre(0)) != 1) then
      Le deuxième chiffre est automatiquement 0, afficher le panneau de limitation de vitesse approprié.
    else
      SVM2.predict(vecteur_chiffre(1));
      Afficher le panneau de limitation de vitesse reconnu;
    end if
  end if
end for
```

---

## 6.6 Résultats expérimentaux

Pour évaluer les performances de notre système, nous utilisons les bases de données GTSRB [97], GTSDDB [92] et BTSD [181] dans les expériences. Pour l'implémentation nous avons utilisé *c++* et la bibliothèque *opencv* fonctionnant sur Intel i5 2.40 GHz et 8 Go de RAM. La base de données GTSDDB utilisée pour tester les étapes de détection et de classification est composée de 159 panneaux circulaires rouges, dont 125 sont des panneaux de limitation de vitesse.

Pour l'évaluation des performances de notre système nous utilisons les deux mé-

## Chapitre 6 : Système de détection et de reconnaissance des panneaux de limitation de vitesse

triques rappel et précision qui sont calculées avec les formules ci-dessous :

$$Précision = \frac{TP}{TP + FP} \quad (6.4)$$

$$Rappel = \frac{TP}{TP + FN} \quad (6.5)$$

Où TP est le nombre de vrais positifs, FP le nombre de faux positifs, et FN le nombre de faux négatifs.

Dans l'étape de détection, la table 6.2 illustre une comparaison entre la méthode de segmentation de lim et al [8] et celle proposée en utilisant Circle Hough Transform (CHT) pour détecter les cercles dans l'image segmentée. Le nombre de faux positifs détectés est fortement réduit de 4000 à 150, ce qui augmente la précision de 3% à 50 %. Le temps de traitement est le même pour les deux méthodes de localisation.

TABLE 6.2 – Comparaison entre la méthode de localisation proposée et la méthode de lim et al. [8].

Méthode	TP	FP	Rappel	Précision	Temps (ms)
Méthode de [8]	152	>4000	95.59%	<3%	28.256
Méthode proposée	151	150	94.96%	50.16%	28.268

Afin d'augmenter la précision, nous proposons une nouvelle méthode basée sur LBP, c'est un filtre qui détecte les régions homogènes dans les images segmentées. Une comparaison entre LBP et la méthode proposée est illustrée dans le tableau 6.3. La méthode proposée est meilleure que la méthode de LBP en terme de rappel et de précision, elle augmente le rappel de 81.76 % à 93.08 % et la précision de 52 % à 64.06 %.

TABLE 6.3 – Comparaison entre LBP et la méthode proposée basée sur LBP.

Méthode	TP	FP	Rappel	Précision	Temps (ms)
LBP	130	120	81.76%	52%	37.524
Méthode proposée	148	83	93.08%	64.06%	37.149

Dans l'étape de classification, dans un premier temps, nous utilisons un seul SVM linéaire avec la transformation HOG, la base de données d'apprentissage est extrait du GTSRB, elle est composé de 11 classes. Les 10 classes positives sont les panneaux de

## Chapitre 6 : Système de détection et de reconnaissance des panneaux de limitation de vitesse

---

limitation de vitesse de 20 à 120 et la classe négative contient d'autres panneaux circulaires rouges. Bien que, nous n'avons pas obtenu de bons résultats, 46.54 % pour le rappel et 30.09 % pour la précision. C'est pourquoi nous avons utilisé deux SVM linéaires avec la transformation HOG. Le premier SVM entraîné sur GTSRB consiste à classer l'objet détecté comme un panneau de limitation de vitesse ou non. Le second SVM entraînée sur la base MNIST, consiste à classer les chiffres des panneaux détectés. Avec cette nouvelle proposition, nous obtenons un rappel de 100 % et une précision de 99.23 % pour le premier SVM qui vérifie si la région détectée est un panneau de limitation de vitesse ou pas, pour le deuxième SVM qui reconnaît les chiffres à l'intérieur des panneaux, nous atteignons 99,63% de rappel et 98,94% de précision respectivement. Par conséquent, nous avons obtenu dans la phase de classification avec la base de données GTSDDB 99,81% de rappel et 99,08% de précision.

Pour évaluer la scalabilité du système proposé, nous l'avons testée sur les bases BTSD et STS, les résultats obtenus sur BTSD sont : 99,03% de rappel et une précision de 98,97% dans le premier SVM, 99,75% de rappel et 99,16% de précision dans le deuxième SVM. Sur la base STS, le système a obtenu un rappel de 98,42% et une précision de 98,38% dans le premier SVM, 98,82% de rappel et de 98,78% de précision dans le deuxième SVM. Donc en moyenne, le système proposé atteint un rappel de classification 99,39% et une précision de 99,05% sur la base BTSD, et sur la base STS nous obtenons un rappel de 98,82% et une précision de 98,78%. Par conséquent, le système proposé est scalable en raison de son architecture et l'utilisation de la base MNIST pour l'apprentissage du deuxième SVM, de sorte que les performances du système sont indépendantes de la base d'apprentissage GTSRB. La précision du deuxième SVM avec BTSD est supérieure à celle avec GTSDDB car la résolution des image de GTSDDB est 1 360x800 pixels et que la taille des images de BTSD est de 1 628x1 230 pixels, donc les chiffres sont clairement capturés et reconnus.

L'avantage du classifieur proposé est non seulement d'augmenter le rappel et la précision, mais aussi de diminuer le temps de traitement en éliminant les faux positifs avec le SVM1. Par conséquent, le système ne perdra pas de temps à traiter les faux positifs avec le SVM2 qui est composé de 9 SVM (un SVM pour chaque chiffre).

Le tableau 6.4 résume les résultats obtenus avec un seul SVM linéaire entraîné uniquement sur la base GTSRB et le système proposé composé de SVM1 et SVM2, nous illustrons les résultats avec les bases de données de test GTSDDB, BTSD et STS. Le système proposé est scalable et rapide en temps de traitement.

Le tableau 6.5 illustre une comparaison entre le système proposé et les approches récentes. Le rappel et la précision de classification obtenus par notre système sont meilleurs que ceux obtenus par les deux méthodes Soetedjo, A. et al [169] et Liu, B. et al [172], ces deux méthodes utilisent la même base de données GTSRB qui est juste une extraction des régions d'intérêt de la base GTSDDB. Soetedjo, A. et al [169] utilisent le template matching correspondant au signe détecté avec un anneau circulaire dans un temps de traitement de 2,95 ms, dans la seconde étape correspondent le nombre de chiffres extrait avec 6 images de référence en 46,78 ms, donc leur temps total de la clas-

## Chapitre 6 : Système de détection et de reconnaissance des panneaux de limitation de vitesse

TABLE 6.4 – Résultats de la classification obtenus par le système proposé.

Méthode	Dataset	Rappel (%)	Précision (%)	Temps (ms)
Un seul SVM entraîné sur GTSRB	GTSDB	46.54	30.09	12.851
Le système proposée	GTSRB	99.81	99.08	11.22
	BTSD	99.39	99.05	
	STS	98.82	98.78	
Le premier SVM	GTSDB	100	99.23	3.845
	BTSD	99.03	98.94	
	STS	98.42	98.38	
SVM2	GTSDB	99.63	98.94	7.387
	BTSD	99.75	99.16	
	STS	99.23	99.18	

sification est de 49,73 ms pour un seul panneau de limitation de vitesse. Cependant, le système proposé prend 3,84 ms dans le premier SVM pour confirmer si la région détectée est un panneau de limitation de vitesse ou pas, et il prend 7,38 ms dans le deuxième SVM pour reconnaître les chiffres, donc au total l'étape de classification atteint 11,22 ms et c'est plus rapide que le temps obtenu par [169]. De plus, notre temps de traitement n'est pas très loin de celui atteint par Gomes, S.L., et al [171] qui atteint un taux de classification de 99.82% dans un temps moyen de 5.595 ms. Les résultats obtenus par notre système proposé sont encore meilleurs que ceux de G. Yan et al [178] (98.51%) et Li, Y. et al [180] (97% sur la base LISA) qui utilisent le CNN,

La comparaison directe de notre système avec les autres méthodes est relativement difficile car les bases de données utilisés dans les tests ne sont pas les mêmes. le système proposé est scalable, il donne les mêmes performances avec différentes base de données BTSD et STS, cependant, les autres méthodes n'ont pas testé leurs systèmes sur des ensembles de données différents.

TABLE 6.5 – Comparaison du système proposé avec les approches de l'état de l'art.

Auteur	Méthode	Année	Dataset	Accuracy	Rappel	Précision	Temps
agudo et al. [166]	SVM	2016	Railway videos	-	95%	-	-
gomes et al. [171]	SVM (linear)	2016	12520 images	99.82%	-	-	5.595 ms
Kundu et al. [168]	Neural network	2015	12300 images	-	98%	-	407 fps
Li et al. [180]	CNN	2016	LISA-TS	97%	-	-	-
Liu et al. [172]	Bag of word	2012	GTSRB	-	97.31%	-	-
Mammeri et al. [167]	SVM	2013	NASLS	-	94%	95%	-
Soetedjo et al. [169]	Template matching	2017	GTSRB	-	93.67%	-	49.73 ms
Yan et al. [178]	CNN	2017	-	98.51%	-	-	-
<b>Système proposé</b>	<b>SVM</b>	<b>2018</b>	<b>GTSDDB</b>	<b>-</b>	<b>99.81%</b>	<b>99.08%</b>	<b>11.22 ms</b>
			<b>BTSD</b>	<b>-</b>	<b>99.39%</b>	<b>99.05%</b>	<b>11.22 ms</b>
			<b>STS</b>	<b>-</b>	<b>98.82%</b>	<b>98.78%</b>	<b>11.22 ms</b>

## **6.7 Conclusion**

Dans ce chapitre, nous avons présenté un système efficace de reconnaissance des panneaux de limitation de vitesse en temps réel. Dans l'étape de détection, nous avons proposé une nouvelle méthode de segmentation qui réduit de manière significative le nombre de fausses alarmes de plus de 4 000 à 150 sur la base BTSD, qui sont éliminées ultérieurement dans l'étape de classification avec le premier SVM. Notre système développé a atteint un rappel de classification élevé (99,81%) et une haute précision (99,08%) dans un temps de traitement de 11,22 ms sur la base de données GTSRB.

L'utilisation de la base de données MNIST pour l'apprentissage rend le système proposé plus scalable et donne presque les mêmes résultats avec d'autres bases de données comme BTSD et STS. Notre système a obtenu un rappel de 99,39% avec BTSD et 98,82% avec STS, et une précision de 99,05% et 98,78% respectivement. Un autre avantage du système est le temps de traitement, où les fausses alarmes détectées dans l'étape de détection sont éliminées avec le premier SVM qui est plus rapide que le second, ainsi le second SVM ne sera pas exécuté seulement si la région détectée est un panneau de limitation de vitesse.

Le système proposé est efficace avec un rappel et une précision élevés, avec une architecture simple, un système scalable qui permet d'obtenir un rappel et une précision élevé sur différentes bases de tests, également un système rapide avec un traitement en temps réel. Le système proposé peut être facilement implémenté sur un calculateur de véhicule ou sur un smartphone qui sera simplement installé sur le tableau de bord de véhicule. Ensuite, le système informera le conducteur de la limitation de vitesse indiquée dans le panneau.

# Conclusion générale

Durant cette thèse, nous nous sommes intéressés au problème de la reconnaissance des panneaux routiers dans les images fixes. Comme conclusion générale nous présentons un résumé des approches proposées, les contributions de la recherche, les résultats obtenus et les perspectives pour des travaux futurs.

La reconnaissance des panneaux routiers est une tâche très importante pour les systèmes d'aide à la conduite. Actuellement, aucune solution optimale au problème de la reconnaissance des panneaux n'est encore connue. Bien que, des progrès importants dans ce domaine aient permis le développement de quelques systèmes pratiques et atteindre un taux de reconnaissance comparable aux performances humaines demeure utopique.

Pour cette raison, ce domaine de recherche constitue encore un terrain fertile pour des futurs travaux. Cette thèse a été consacrée à l'étude et la réalisation d'un système de reconnaissance des panneaux routiers.

## Contributions

Nous rappelons ici les principales contributions de ce manuscrit. Elles sont organisées en trois axes : La segmentation, la détection et la classification. Nous présentons ainsi le système de reconnaissance des panneaux de limitation de vitesse développé et les résultats obtenus

**La segmentation des images :** C'est est une étape cruciale dans un système de reconnaissance des panneaux routiers, elle consiste à préparer l'image afin de la rendre plus exploitable par les processus des étapes suivantes. Dans cette thèse, nous avons comparé plusieurs méthodes de segmentation dans différents espaces de couleurs et nous avons choisi l'espace RGB normalisé car il donne de bons résultats en termes de performance et temps d'exécution.

Pour localiser les régions d'intérêt, nous avons proposé une nouvelle méthode de segmentation basée sur le modèle Itti qui réduit significativement le nombre des faux positifs, ce qui augmente considérablement la précision de 3% jusqu'à plus de 50%.

## Conclusion générale

---

**Détection des panneaux routiers** : La détection des panneaux routiers assume un rôle crucial car si le panneau n'est pas bien localisé et détecté, il sera mal classifié dans la phase de classification. La deuxième contribution est la proposition d'une nouvelle approche basée sur LBP qui sert à détecter les régions homogènes dans les images segmentées afin d'éliminer plus de faux positifs. Avec la méthode proposée la précision est augmentée de 50% à 64%.

**Classification des panneaux routiers** : C'est l'étape principale du système de reconnaissance des panneaux routier où les panneaux détectés sont reconnus et identifiés. Nous avons testé plusieurs méthodes pour la reconnaissance des panneaux routiers : les méthodes des points d'intérêt SIFT et SURF, le template matching et le classifieur SVM et nous avons adopté SVM en raison de ses performances et son temps de traitement. Nous avons utilisé le descripteur HOG avec le SVM due à sa rapidité et efficacité.

**Système de reconnaissance des panneaux de limitation de vitesse** : Notre principale contribution dans cette thèse est le développement d'un système de reconnaissance des panneaux de limitation de vitesse en utilisons les approches proposées pour la détection des panneaux. Dans la phase de classification, nous avons utilisé une architecture composée de deux étapes. Dans la première, un classifieur SVM avec un noyau linéaire entraîné sur la base GTSRB est utilisé pour reconnaître si la région détectée est un panneaux de limitation de vitesse ou non. Dans la deuxième étape, nous avons utilisé un deuxième classifieur SVM avec un noyau linéaire entraîné sur la base MNIST pour classifier les chiffres du panneau détecté. Avec la base GTSRB le système atteint un rappel 99.81% et une précision 99.08% dans un temps 11.22 ms. Le système proposé est non seulement performant mais aussi scalable en raison de l'utilisation de la base de données pour l'apprentissage, avec la base BTSD le système atteint 99.39% de rappel et 99.05% de précision. Nous avons aussi évalué le système développé sur la base STS et il atteint un rappel de 98.82% et une précision de 98.78%. Une comparaison des performances entre le système proposé et les méthodes récentes de l'état de l'art montre l'efficacité de notre système avec un rappel et une précision élevés. En plus de ces bonnes performances, l'architecture simple et la scalabilité du système lui permettent d'obtenir des résultats optimaux sur différentes bases de tests et avec un traitement en temps réel.

## Perspectives

Les résultats obtenus pendant ces années ouvrent un certain nombre de perspectives pour des travaux futurs.

- Créer une nouvelle base de données des panneaux routiers plus complexe et universelle. Une base qui contient des images capturées durant le jour et la nuit, ainsi que pendant toutes les saisons de l'année.
- Utiliser le R-CNN pour la localisation et la détection des panneaux routiers pour

## Conclusion générale

---

avoir un taux de détection élevé.

- La reconnaissance des panneaux de priorité pour développer un système de décision de priorité selon les panneaux détectés, les véhicules présentes dans l'intersection et les règles de priorités apprises.

Nous envisageons l'implémentation de notre système sur un smart phone afin qu'il soit facile à utiliser par tous les conducteurs.

# Bibliographie

- [1] Arturo De la Escalera, J Ma Armingol, and Mario Mata. Traffic sign recognition and analysis for intelligent vehicles. *Image and vision computing*, 21(3) :247–258, 2003.
- [2] Andrzej Ruta, Yongmin Li, and Xiaohui Liu. Real-time traffic sign recognition from video by class-specific discriminative features. *Pattern Recognition*, 43(1) :416–430, 2010.
- [3] Fei Qin, Bin Fang, and Hengjun Zhao. Traffic sign segmentation and recognition in scene images. In *Pattern Recognition (CCPR), 2010 Chinese Conference on*, pages 1–5. IEEE, 2010.
- [4] Ali Behloul and Yassmina Saadna. A fast and robust traffic sign recognition. *International Journal of Innovation and Applied Studies*, 5(2) :139, 2014.
- [5] Laurent Itti, Christof Koch, and Ernst Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on pattern analysis and machine intelligence*, 20(11) :1254–1259, 1998.
- [6] Pavel Yakimov. Traffic signs detection using tracking with prediction. In *International Conference on E-Business and Telecommunications*, pages 454–467. Springer, 2015.
- [7] Zhixie Chen, Jing Yang, and Bin Kong. A robust traffic sign recognition system for intelligent vehicles. In *Image and Graphics (ICIG), 2011 Sixth International Conference on*, pages 975–980. IEEE, 2011.
- [8] King Hann Lim, Li-Minn Ang, and Kah Phooi Seng. New hybrid technique for traffic sign recognition. In *Intelligent Signal Processing and Communications Systems, 2008. ISPACS 2008. International Symposium on*, pages 1–4. IEEE, 2009.
- [9] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2) :91–110, 2004.
- [10] Manuel Grand-Brochier. *Descripteurs 2D et 2D+ t de points d'intérêt pour des appariements robustes*. PhD thesis, Université Blaise Pascal-Clermont-Ferrand II, 2011.
- [11] VS Hakim. Implementation and analysis of real-time object tracking on the starburst mpsoc. Master's thesis, University of Twente, 2015.
- [12] Ali Youssef, Dario Albani, Daniele Nardi, and Domenico Daniele Bloisi. Fast traffic sign recognition using color segmentation and deep convolutional networks.

## Conclusion générale

---

- In *International Conference on Advanced Concepts for Intelligent Vision Systems*, pages 205–216. Springer, 2016.
- [13] PY Yakimov. Real-time road signs recognition using mobile gpu. In *CEUR workshop proceedings*, volume 1638, pages 477–483, 2016.
- [14] Zumra Malik and Imran Siddiqi. Detection and recognition of traffic signs from road scene images. In *Frontiers of Information Technology (FIT), 2014 12th International Conference on*, pages 330–335. IEEE, 2014.
- [15] Hamed Habibi Aghdam, Elnaz Jahani Heravi, and Domenec Puig. A practical approach for detection and classification of traffic signs using convolutional neural networks. *Robotics and autonomous systems*, 84 :97–112, 2016.
- [16] Junqi Jin, Kun Fu, and Changshui Zhang. Traffic sign recognition with hinge loss trained convolutional neural networks. *IEEE Transactions on Intelligent Transportation Systems*, 15(5) :1991–2000, 2014.
- [17] Bahman Soheilian, Aurore Arlicot, and Nicolas Papanoditis. Extraction de panneaux de signalisation routière dans des images couleurs. In *Reconnaissance des Formes et Intelligence Artificielle*, pages 1–8, 2010.
- [18] Michael Shneier. Road sign detection and recognition. In *IEEE Computer Society International Conference on Computer Vision and Pattern Recognition*. IEEE, June, 2005.
- [19] Jim Torresen, Jorgen W Bakke, and Lukas Sekanina. Efficient recognition of speed limit signs. In *Intelligent Transportation Systems, 2004. Proceedings. The 7th International IEEE Conference on*, pages 652–656. IEEE, 2004.
- [20] Mohamed Benallal and Jean Meunier. Real-time color segmentation of road signs. In *Electrical and Computer Engineering, 2003. IEEE CCECE 2003. Canadian Conference on*, volume 3, pages 1823–1826. IEEE, 2003.
- [21] Vavilin Andrey and Kang Hyun Jo. Automatic detection and recognition of traffic signs using geometric structure analysis. In *SICE-ICASE, 2006. International Joint Conference*, pages 1451–1456. IEEE, 2006.
- [22] Pavel Pačlık, J Novovičová, Pavel Pudil, and Petr Somol. Road sign classification using laplace kernel classifier. *Pattern Recognition Letters*, 21(13) :1165–1173, 2000.
- [23] Hasan Fleyeh. Shadow and highlight invariant colour segmentation algorithm for traffic signs. In *Cybernetics and Intelligent Systems, 2006 IEEE Conference on*, pages 1–7. IEEE, 2006.
- [24] Salvatore Vitabile, Giorgio Pollaccia, Giovanni Pilato, and Filippo Sorbello. Road signs recognition using a dynamic pixel aggregation technique in the hsv color space. In *Image Analysis and Processing, 2001. Proceedings. 11th International Conference on*, pages 572–577. IEEE, 2001.
- [25] Paolo Medici, Claudio Caraffi, Elena Cardarelli, Pier Paolo Porta, and Guido Ghisio. Real time road signs classification. In *Vehicular Electronics and Safety, 2008. ICVES 2008. IEEE International Conference on*, pages 253–258. IEEE, 2008.

## Conclusion générale

---

- [26] Wen-Yen Wu, Tsung-Cheng Hsieh, and Ching-Sung Lai. Extracting road signs using the color information. *World Academy of Science, Engineering and Technology*, 32 :282–286, 2007.
- [27] U Zakir, ANJ Leonce, and EA Edirisinghe. Road sign segmentation based on colour spaces : A comparative study. In *Proceedings of the 11th Iasted International Conference on Computer Graphics and Imgaing, Innsbruck, Austria, 2010*.
- [28] Gaurav Sharma and H Joel Trussell. Digital color imaging. *IEEE transactions on image processing*, 6(7) :901–932, 1997.
- [29] Jun Miura, Tsuyoshi Kanda, Shusaku Nakatani, and Yoshiaki Shirai. An active vision system for on-line traffic sign recognition. *IEICE TRANSACTIONS on Information and Systems*, 85(11) :1784–1792, 2002.
- [30] Alberto Broggi, Pietro Cerri, Paolo Medici, Pier Paolo Porta, and Guido Ghisio. Real time road signs recognition. In *Intelligent Vehicles Symposium, 2007 IEEE*, pages 981–986. IEEE, 2007.
- [31] Ahmed Hechri, Rihab Hmida, and Abdellatif Mtibaa. Robust road lanes and traffic signs recognition for driver assistance system. *International Journal of Computational Science and Engineering*, 10(1-2) :202–209, 2015.
- [32] Miguel Angel Garcia-Garrido, Miguel Angel Sotelo, and Ernesto Martin-Gorostiza. Fast traffic sign detection and recognition under changing lighting conditions. In *Intelligent Transportation Systems Conference, 2006. ITSC'06. IEEE*, pages 811–816. IEEE, 2006.
- [33] Nick Barnes and Alex Zelinsky. Real-time radial symmetry for speed sign detection. In *Intelligent Vehicles Symposium, 2004 IEEE*, pages 566–571. IEEE, 2004.
- [34] Gareth Loy and Nick Barnes. Fast shape-based road sign detection for a driver assistance system. In *Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, volume 1, pages 70–75. IEEE, 2004.
- [35] Paul Viola and 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–I. IEEE, 2001.
- [36] Karla Brkic, Axel Pinz, and Siniša Šegvic. Traffic sign detection as a component of an automated traffic infrastructure inventory system. *Stainz, Austria, May, 2009*.
- [37] Karla Brkić, Siniša Šegvić, Zoran Kalafatić, Ivan Sikirić, and Axel Pinz. Generative modeling of spatio-temporal traffic sign trajectories. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*, pages 25–31. IEEE, 2010.
- [38] Tao Chen and Shijian Lu. Accurate and efficient traffic sign detection using discriminative adaboost and support vector regression. *IEEE Transactions on Vehicular Technology*, 65(6) :4006–4015, 2016.
- [39] Victor Adrian Prisacariu, Radu Timofte, Karel Zimmermann, Ian Reid, and Luc Van Gool. Integrating object detection with 3d tracking towards a better driver

- assistance system. In *Pattern Recognition (ICPR), 2010 20th International Conference on*, pages 3344–3347. IEEE, 2010.
- [40] Chiung-Yao Fang, Sei-Wang Chen, and Chiou-Shann Fuh. Road-sign detection and tracking. *IEEE transactions on vehicular technology*, 52(5) :1329–1341, 2003.
- [41] Yihui Wu, Yulong Liu, Jianmin Li, Huaping Liu, and Xiaolin Hu. Traffic sign detection based on convolutional neural networks. In *Neural Networks (IJCNN), The 2013 International Joint Conference on*, pages 1–7. IEEE, 2013.
- [42] Di Zang, Junqi Zhang, Dongdong Zhang, Maomao Bao, Jiujun Cheng, and Keshuang Tang. Traffic sign detection based on cascaded convolutional neural networks. In *Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), 2016 17th IEEE/ACIS International Conference on*, pages 201–206. IEEE, 2016.
- [43] Fatin Zaklouta, Bogdan Stanculescu, and 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.
- [44] Ayoub Ellahyani, Mohamed El Ansari, and Ilyas El Jaafari. Traffic sign detection and recognition based on random forests. *Applied Soft Computing*, 46 :805–815, 2016.
- [45] Chen Li and Cheng Yang. The research on traffic sign recognition based on deep learning. In *Communications and Information Technologies (ISCIT), 2016 16th International Symposium on*, pages 156–161. IEEE, 2016.
- [46] Benjamin Hoferlin and Klaus Zimmermann. Towards reliable traffic sign recognition. In *Intelligent Vehicles Symposium, 2009 IEEE*, pages 324–329. IEEE, 2009.
- [47] P Sathish and D Bharathi. Automatic road sign detection and recognition based on sift feature matching algorithm. In *Proceedings of the International Conference on Soft Computing Systems*, pages 421–431. Springer, 2016.
- [48] Xiaoguang HUA, Xinyan ZHUA, Deren LIA, and Hui LI. Traffic sign recognition using scale invariant feature transform and svm. In *A special joint symposium of ISPRS technical commission IV and AutoCarto in conjunction with ASPRS/CaGIS fall specialty conference November*, pages 15–19, 2010.
- [49] Md Zainal Abedin, Prashengit Dhar, and Kaushik Deb. Traffic sign recognition using surf : speeded up robust feature descriptor and artificial neural network classifier. In *Electrical and Computer Engineering (ICECE), 2016 9th International Conference on*, pages 198–201. IEEE, 2016.
- [50] Yan Han, Kushal Virupakshappa, and Erdal Oruklu. Robust traffic sign recognition with feature extraction and k-nn classification methods. In *Electro/Information Technology (EIT), 2015 IEEE International Conference on*, pages 484–488. IEEE, 2015.
- [51] Monika Lasota and Marcin Skoczylas. Recognition of multiple traffic signs using keypoints feature detectors. In *Electrical and Power Engineering (EPE), 2016 International Conference and Exposition on*, pages 535–540. IEEE, 2016.

- [52] Stefan Eickeler, Matias Valdenegro, Thomas Werner, and Michael Kieninger. Future computer vision algorithms for traffic sign recognition systems. In *Advanced Microsystems for Automotive Applications 2015*, pages 69–77. Springer, 2016.
- [53] Dan Cireşan, Ueli Meier, Jonathan Masci, and Jürgen Schmidhuber. A committee of neural networks for traffic sign classification. In *Neural Networks (IJCNN), The 2011 International Joint Conference on*, pages 1918–1921. IEEE, 2011.
- [54] Dan CireşAn, Ueli Meier, Jonathan Masci, and Jürgen Schmidhuber. Multi-column deep neural network for traffic sign classification. *Neural Networks*, 32 :333–338, 2012.
- [55] Pierre Sermanet and Yann LeCun. Traffic sign recognition with multi-scale convolutional networks. In *Neural Networks (IJCNN), The 2011 International Joint Conference on*, pages 2809–2813. IEEE, 2011.
- [56] Rongqiang Qian, Yong Yue, Frans Coenen, and Bailing Zhang. Traffic sign recognition with convolutional neural network based on max pooling positions. In *Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD), 2016 12th International Conference on*, pages 578–582. IEEE, 2016.
- [57] Kaixuan Xie, Shiming Ge, Qiting Ye, and Zhao Luo. Traffic sign recognition based on attribute-refinement cascaded convolutional neural networks. In *Pacific Rim Conference on Multimedia*, pages 201–210. Springer, 2016.
- [58] Yassmina Saadna and Ali Behloul. An overview of traffic sign detection and classification methods. *International Journal of Multimedia Information Retrieval*, 6(3) :193–210, 2017.
- [59] Arturo De La Escalera, Luis E Moreno, Miguel Angel Salichs, and José María Armingol. Road traffic sign detection and classification. *IEEE transactions on industrial electronics*, 44(6) :848–859, 1997.
- [60] Jianping Wu and Yichang James Tsai. Enhanced roadway inventory using a 2-d sign video image recognition algorithm. *Computer-Aided Civil and Infrastructure Engineering*, 21(5) :369–382, 2006.
- [61] Tam T Le, Son T Tran, Seichii Mita, and Thuc D Nguyen. Real time traffic sign detection using color and shape-based features. In *Asian Conference on Intelligent Information and Database Systems*, pages 268–278. Springer, 2010.
- [62] King Hann Lim, Kah Phooi Seng, and Li Minn Ang. Intra color-shape classification for traffic sign recognition. In *Computer Symposium (ICS), 2010 International*, pages 642–647. IEEE, 2010.
- [63] Saturnino Maldonado-Bascon, Sergio Lafuente-Arroyo, Pedro Gil-Jimenez, Hilario Gomez-Moreno, and Francisco López-Ferreras. Road-sign detection and recognition based on support vector machines. *IEEE transactions on intelligent transportation systems*, 8(2) :264–278, 2007.
- [64] Andrzej Ruta, Fatih Porikli, Shintaro Watanabe, and Yongmin Li. In-vehicle camera traffic sign detection and recognition. *Machine Vision and Applications*, 22(2) :359–375, 2011.

- [65] Kenji Hirose, Toshiyuki Asakura, and Yuji Aoyagi. Real-time recognition of road traffic sign in moving scene image using new image filter. In *Industrial Electronics Society, 2000. IECON 2000. 26th Annual Conference of the IEEE*, volume 3, pages 2207–2212. IEEE, 2000.
- [66] Shuangdong Zhu and Lanlan Liu. Traffic sign recognition based on color standardization. In *Information Acquisition, 2006 IEEE International Conference on*, pages 951–955. IEEE, 2006.
- [67] Hirofumi Ohara, Ikuko Nishikawa, S Miki, and N Yabuki. Detection and recognition of road signs using simple layered neural networks. In *Neural Information Processing, 2002. ICONIP'02. Proceedings of the 9th International Conference on*, volume 2, pages 626–630. IEEE, 2002.
- [68] Nobuyuki Otsu. A threshold selection method from gray-level histograms. *IEEE transactions on systems, man, and cybernetics*, 9(1) :62–66, 1979.
- [69] P Yu Yakimov. Preprocessing digital images for quickly and reliably detecting road signs. *Pattern Recognition and Image Analysis*, 25(4) :729–732, 2015.
- [70] Gangyi Wang, Guanghui Ren, Lihui Jiang, and Taifan Quan. Hole-based traffic sign detection method for traffic signs with red rim. *The Visual Computer*, 30(5) :539–551, 2014.
- [71] Wen-Jia Kuo and Chien-Chung Lin. Two-stage road sign detection and recognition. In *Multimedia and Expo, 2007 IEEE International Conference on*, pages 1427–1430. IEEE, 2007.
- [72] Yok-Yen Nguwi and Abbas Z Kouzani. Detection and classification of road signs in natural environments. *Neural computing and applications*, 17(3) :265–289, 2008.
- [73] Xu Qingsong, Su Juan, and Liu Tiantian. A detection and recognition method for prohibition traffic signs. In *Image Analysis and Signal Processing (IASP), 2010 International Conference on*, pages 583–586. IEEE, 2010.
- [74] Chiung-Yao Fang, Chiou-Shann Fuh, PS Yen, Shen Cherng, and Sei-Wang Chen. An automatic road sign recognition system based on a computational model of human recognition processing. *Computer vision and Image understanding*, 96(2) :237–268, 2004.
- [75] Michaël Freylinger and Jean-François Verdin. *Détection et reconnaissance de signaux routiers dans un flux vidéo*. PhD thesis, Institut Montefiore, Université de Liège, Belgique., 2005.
- [76] Rachid Belaroussi and Jean Philippe Tarel. Détection des panneaux de signalisation routière par accumulation bivariée. *Traitement du signal*, 27(3) :pp–265, 2010.
- [77] Hilario Gómez-Moreno, Saturnino Maldonado-Bascón, Pedro Gil-Jiménez, and Sergio Lafuente-Arroyo. Goal evaluation of segmentation algorithms for traffic sign recognition. *IEEE Transactions on Intelligent Transportation Systems*, 11(4) :917–930, 2010.

- [78] Chunsheng Liu, Faliang Chang, Zhenxue Chen, and Dongmei Liu. Fast traffic sign recognition via high-contrast region extraction and extended sparse representation. *IEEE transactions on Intelligent transportation systems*, 17(1) :79–92, 2016.
- [79] Chunsheng Liu, Faliang Chang, and Zhenxue Chen. Rapid multiclass traffic sign detection in high-resolution images. *IEEE Transactions on Intelligent Transportation Systems*, 15(6) :2394–2403, 2014.
- [80] Gary R Bradski. Computer vision face tracking for use in a perceptual user interface. *Intel Technology Journal Q2*, pages 705–740, 1998.
- [81] Ge Song and Hong Wang. A fast and robust ellipse detection algorithm based on pseudo-random sample consensus. In *Computer Analysis of Images and Patterns*, pages 669–676. Springer, 2007.
- [82] Si-Cheng Zhang and Zhi-Qiang Liu. A robust, real-time ellipse detector. *Pattern Recognition*, 38(2) :273–287, 2005.
- [83] Fredrik Larsson, Michael Felsberg, et al. Using fourier descriptors and spatial models for traffic sign recognition. In *SCIA*, volume 11, pages 238–249. Springer, 2011.
- [84] S Lafuente-Arroyo, P Gil-Jimenez, R Maldonado-Bascon, F Lopez-Ferreras, and S Maldonado-Bascon. Traffic sign shape classification evaluation i : Svm using distance to borders. In *Intelligent Vehicles Symposium, 2005. Proceedings. IEEE*, pages 557–562. IEEE, 2005.
- [85] Cuneyt Akinlar and Cihan Topal. Edcircles : A real-time circle detector with a false detection control. *Pattern Recognition*, 46(3) :725–740, 2013.
- [86] Selcan Kaplan Berkaya, Huseyin Gunduz, Ozgur Ozsen, Cuneyt Akinlar, and Serkan Gunal. On circular traffic sign detection and recognition. *Expert Systems with Applications*, 48 :67–75, 2016.
- [87] Karla Brkic. An overview of traffic sign detection methods. *Department of Electronics, Microelectronics, Computer and Intelligent Systems Faculty of Electrical Engineering and Computing Unska*, 3 :10000, 2010.
- [88] Xavier Baró, Sergio Escalera, Jordi Vitrià, Oriol Pujol, and Petia Radeva. Traffic sign recognition using evolutionary adaboost detection and forest-ecoc classification. *IEEE Transactions on Intelligent Transportation Systems*, 10(1) :113–126, 2009.
- [89] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886–893. IEEE, 2005.
- [90] Fatin Zaklouta and Bogdan Stanciulescu. Warning traffic sign recognition using a hog-based kd tree. In *Intelligent Vehicles Symposium (IV), 2011 IEEE*, pages 1019–1024. IEEE, 2011.
- [91] Gangyi Wang, Guanghui Ren, Zhilu Wu, Yaqin Zhao, and Lihui Jiang. A robust, coarse-to-fine traffic sign detection method. In *Neural Networks (IJCNN), The 2013 International Joint Conference on*, pages 1–5. IEEE, 2013.

## Conclusion générale

---

- [92] Sebastian Houben, Johannes Stallkamp, Jan Salmen, Marc Schlipsing, and Christian Igel. Detection of traffic signs in real-world images : The german traffic sign detection benchmark. In *Neural Networks (IJCNN), The 2013 International Joint Conference on*, pages 1–8. IEEE, 2013.
- [93] Sebastian Houben. A single target voting scheme for traffic sign detection. In *Intelligent Vehicles Symposium (IV), 2011 IEEE*, pages 124–129. IEEE, 2011.
- [94] Samuele Salti, Alioscia Petrelli, Federico Tombari, Nicola Fioraio, and Luigi Di Stefano. Traffic sign detection via interest region extraction. *Pattern Recognition*, 48(4) :1039–1049, 2015.
- [95] Markus Mathias, Radu Timofte, Rodrigo Benenson, and Luc Van Gool. Traffic sign recognition—how far are we from the solution? In *Neural Networks (IJCNN), The 2013 International Joint Conference on*, pages 1–8. IEEE, 2013.
- [96] Chunsheng Liu, Faliang Chang, Zhenxue Chen, and Shuang Li. Rapid traffic sign detection and classification using categories-first-assigned tree. 2013.
- [97] Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. Man vs. computer : Benchmarking machine learning algorithms for traffic sign recognition. *Neural networks*, 32 :323–332, 2012.
- [98] Yanyun Qu, Siying Yang, Weiwei Wu, and Li Lin. Hierarchical traffic sign recognition. In *Pacific Rim Conference on Multimedia*, pages 200–209. Springer, 2016.
- [99] Huaping Liu, Yulong Liu, and Fuchun Sun. Traffic sign recognition using group sparse coding. *Information Sciences*, 266 :75–89, 2014.
- [100] Long Chen, Qingquan Li, Ming Li, and Qingzhou Mao. Traffic sign detection and recognition for intelligent vehicle. In *Intelligent Vehicles Symposium (IV), 2011 IEEE*, pages 908–913. IEEE, 2011.
- [101] Hamed Habibi Aghdam, Elnaz Jahani Heravi, and Domenec Puig. Traffic sign recognition using visual attributes and bayesian network. In *International Joint Conference on Computer Vision, Imaging and Computer Graphics*, pages 295–315. Springer, 2015.
- [102] Xiaojun He and Benqi Dai. A new traffic signs classification approach based on local and global features extraction. In *Information Communication and Management (ICICM), International Conference on*, pages 121–125. IEEE, 2016.
- [103] Suisui Tang and Lin-Lin Huang. Traffic sign recognition using complementary features. In *Pattern Recognition (ACPR), 2013 2nd IAPR Asian Conference on*, pages 210–214. IEEE, 2013.
- [104] Yujun Zeng, Xin Xu, Yuqiang Fang, and Kun Zhao. Traffic sign recognition using extreme learning classifier with deep convolutional features. In *The 2015 international conference on intelligence science and big data engineering (IScIDE 2015), Suzhou, China*, volume 9242, pages 272–280, 2015.
- [105] Dan Ciregan, Ueli Meier, and Jürgen Schmidhuber. Multi-column deep neural networks for image classification. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3642–3649. IEEE, 2012.

## Conclusion générale

---

- [106] Hamed Habibi Aghdam, Elnaz Jahani Heravi, and Domenec Puig. A practical and highly optimized convolutional neural network for classifying traffic signs in real-time. *International Journal of Computer Vision*, 122(2) :246–269, 2017.
- [107] Hamed H Aghdam, Elnaz J Heravi, and Domenec Puig. Recognizing traffic signs using a practical deep neural network. In *Robot 2015 : Second Iberian Robotics Conference*, pages 399–410. Springer, 2016.
- [108] Steven L Horowitz. Picture segmentation by a directed split-and-merge procedure. In *IJCPR*, pages 424–433, 1974.
- [109] Niceto Salazar. *Advanced Concepts in Image Segmentation*. Clanrye International, 2015.
- [110] King-Sun Fu and JK Mui. A survey on image segmentation. *Pattern recognition*, 13(1) :3–16, 1981.
- [111] CK Chow and T Kaneko. Boundary detection of radiographic images by a threshold method. In *Frontiers of pattern recognition*, pages 61–82. Elsevier, 1972.
- [112] John A Hartigan and Manchek A Wong. Algorithm as 136 : A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1) :100–108, 1979.
- [113] James MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA, 1967.
- [114] Youji Fukada. Spatial clustering procedures for region analysis. *Pattern Recognition*, 12(6) :395–403, 1980.
- [115] Lawrence G Roberts. *Machine perception of three-dimensional solids*. PhD thesis, Massachusetts Institute of Technology, 1963.
- [116] Irwin Sobel. Camera models and machine perception. Technical report, Stanford Univ Calif Dept of Computer Science, 1970.
- [117] John Canny. A computational approach to edge detection. In *Readings in Computer Vision*, pages 184–203. Elsevier, 1987.
- [118] Rachid Deriche. Using canny’s criteria to derive a recursively implemented optimal edge detector. *International journal of computer vision*, 1(2) :167–187, 1987.
- [119] Randy Crane. *Simplified approach to image processing : classical and modern techniques in C*. Prentice Hall PTR, 1996.
- [120] Donald Hearn and M Pauline Baker. Computer graphics, c version. *Prentice Hall Inc., New Jersey, US*, 87 :88, 1997.
- [121] Andreas Koschan and Mongi Abidi. *Digital color image processing*. John Wiley & Sons, 2008.
- [122] Simone Frintrop, Gerriet Backer, and Erich Rome. Goal-directed search with a top-down modulated computational attention system. In *Joint Pattern Recognition Symposium*, pages 117–124. Springer, 2005.

## Conclusion générale

---

- [123] Dirk Walther, Ueli Rutishauser, Christof Koch, and Pietro Perona. Selective visual attention enables learning and recognition of multiple objects in cluttered scenes. *Computer Vision and Image Understanding*, 100(1-2) :41–63, 2005.
- [124] Zhicheng Li and Laurent Itti. Saliency and gist features for target detection in satellite images. *IEEE Transactions on Image Processing*, 20(7) :2017–2029, 2011.
- [125] Zhenghu Ding, Ying Yu, Bin Wang, and Liming Zhang. An approach for visual attention based on biquaternion and its application for ship detection in multispectral imagery. *Neurocomputing*, 76(1) :9–17, 2012.
- [126] Ali Borji and Laurent Itti. State-of-the-art in visual attention modeling. *IEEE transactions on pattern analysis and machine intelligence*, 35(1) :185–207, 2013.
- [127] Olivier Le Meur, Patrick Le Callet, Dominique Barba, and Dominique Thoreau. A coherent computational approach to model bottom-up visual attention. *IEEE transactions on pattern analysis and machine intelligence*, 28(5) :802–817, 2006.
- [128] Olivier Le Meur, Patrick Le Callet, and Dominique Barba. Predicting visual fixations on video based on low-level visual features. *Vision research*, 47(19) :2483–2498, 2007.
- [129] Vidhya Navalpakkam and Laurent Itti. An integrated model of top-down and bottom-up attention for optimizing detection speed. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 2049–2056. IEEE, 2006.
- [130] Naila Murray, Maria Vanrell, Xavier Otazu, and C Alejandro Parraga. Saliency estimation using a non-parametric low-level vision model. In *Computer vision and pattern recognition (cvpr), 2011 ieee conference on*, pages 433–440. IEEE, 2011.
- [131] Sophie Marat, Tien Ho Phuoc, Lionel Granjon, Nathalie Guyader, Denis Pellerin, and Anne Guérin-Dugué. Modelling spatio-temporal saliency to predict gaze direction for short videos. *International journal of computer vision*, 82(3) :231, 2009.
- [132] Antonio Torralba. Modeling global scene factors in attention. *JOSA A*, 20(7) :1407–1418, 2003.
- [133] Aude Oliva, Antonio Torralba, Monica S Castelhana, and John M Henderson. Top-down control of visual attention in object detection. In *Image processing, 2003. icip 2003. proceedings. 2003 international conference on*, volume 1, pages I–253. IEEE, 2003.
- [134] Krista A Ehinger, Barbara Hidalgo-Sotelo, Antonio Torralba, and Aude Oliva. Modelling search for people in 900 scenes : A combined source model of eye guidance. *Visual cognition*, 17(6-7) :945–978, 2009.
- [135] Laurent Itti and Pierre F Baldi. Bayesian surprise attracts human attention. In *Advances in neural information processing systems*, pages 547–554, 2006.
- [136] Dashan Gao and Nuno Vasconcelos. Discriminant saliency for visual recognition from cluttered scenes. In *Advances in neural information processing systems*, pages 481–488, 2005.

## Conclusion générale

---

- [137] Vijay Mahadevan and Nuno Vasconcelos. Spatiotemporal saliency in dynamic scenes. *IEEE transactions on pattern analysis and machine intelligence*, 32(1) :171–177, 2010.
- [138] Christof Koch and Shimon Ullman. Shifts in selective visual attention : towards the underlying neural circuitry. In *Matters of intelligence*, pages 115–141. Springer, 1987.
- [139] Ruggero Milanese, Sylvia Gil Milanese, and Thierry Pun. Attentive mechanisms for dynamic and static scene analysis. *Optical engineering*, 34(8) :2428–2434, 1995.
- [140] Shumeet Baluja and Dean A Pomerleau. Expectation-based selective attention for visual monitoring and control of a robot vehicle. *Robotics and autonomous systems*, 22(3-4) :329–344, 1997.
- [141] Anne M Treisman and Garry Gelade. A feature-integration theory of attention. *Cognitive psychology*, 12(1) :97–136, 1980.
- [142] Stephen Engel, Xuemei Zhang, and Brian Wandell. Colour tuning in human visual cortex measured with functional magnetic resonance imaging. *Nature*, 388(6637) :68, 1997.
- [143] Paul VC Hough. Method and means for recognizing complex patterns, December 18 1962. US Patent 3,069,654.
- [144] Violet F Leavers. *Shape detection in computer vision using the Hough transform*. Springer, 1992.
- [145] Harvey Rhody. Lecture 10 : Hough circle transform. *Chester F Carlson Center for Imaging Science, Rochester Institute of Technology*, 2005.
- [146] Timo Ojala, Matti Pietikäinen, and David Harwood. A comparative study of texture measures with classification based on featured distributions. *Pattern recognition*, 29(1) :51–59, 1996.
- [147] Matti Pietikäinen, Abdenour Hadid, Guoying Zhao, and Timo Ahonen. *Computer vision using local binary patterns*, volume 40. Springer Science & Business Media, 2011.
- [148] Timo Ojala, Matti Pietikainen, and Topi Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on pattern analysis and machine intelligence*, 24(7) :971–987, 2002.
- [149] Matti Pietikäinen, Timo Ojala, and Zelin Xu. Rotation-invariant texture classification using feature distributions. *Pattern Recognition*, 33(1) :43–52, 2000.
- [150] Timo Ahonen, Abdenour Hadid, and Matti Pietikainen. Face description with local binary patterns : Application to face recognition. *IEEE transactions on pattern analysis and machine intelligence*, 28(12) :2037–2041, 2006.
- [151] Caifeng Shan and Tommaso Gritti. Learning discriminative lbp-histogram bins for facial expression recognition. In *BMVC*, pages 1–10, 2008.
- [152] Ramin Zabih and John Woodfill. Non-parametric local transforms for computing visual correspondence. In *European conference on computer vision*, pages 151–158. Springer, 1994.

## Conclusion générale

---

- [153] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (surf). *Computer vision and image understanding*, 110(3) :346–359, 2008.
- [154] Nicolas Noury. *Mise en correspondance A Contrario de points d'intérêt sous contraintes géométrique et photométrique*. PhD thesis, Université Henri Poincaré-Nancy I, 2011.
- [155] Marius Muja and David G Lowe. Fast approximate nearest neighbors with automatic algorithm configuration. *VISAPP (1)*, 2(331-340) :2, 2009.
- [156] Chanop Silpa-Anan and Richard Hartley. Optimised kd-trees for fast image descriptor matching. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.
- [157] Roberto Brunelli. *Template matching techniques in computer vision : theory and practice*. John Wiley & Sons, 2009.
- [158] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3) :273–297, 1995.
- [159] Shigeo Abe. *Support vector machines for pattern classification*, volume 2. Springer, 2005.
- [160] Lipo Wang. *Support vector machines : theory and applications*, volume 177. Springer Science & Business Media, 2005.
- [161] Christopher JC Burges. A tutorial on support vector machines for pattern recognition. *Data mining and knowledge discovery*, 2(2) :121–167, 1998.
- [162] Nello Cristianini and John Shawe-Taylor. *An introduction to support vector machines and other kernel-based learning methods*. Cambridge university press, 2000.
- [163] Vojislav Kecman. *Learning and soft computing : support vector machines, neural networks, and fuzzy logic models*. MIT press, 2001.
- [164] Hilton Bristow and Simon Lucey. Why do linear svms trained on hog features perform so well? *arXiv preprint arXiv :1406.2419*, 2014.
- [165] Anne-Sophie Puthon. *Détermination de la vitesse limite par fusion de données vision et cartographiques temps-réel embarquées*. PhD thesis, Ecole Nationale Supérieure des Mines de Paris, 2013.
- [166] David Agudo, Ángel Sánchez, José F Vélez, and A Belén Moreno. Real-time railway speed limit sign recognition from video sequences. In *Systems, Signals and Image Processing (IWSSIP), 2016 International Conference on*, pages 1–4. IEEE, 2016.
- [167] Abdelhamid Mammeri, Azzedine Boukerche, Jingwen Feng, and Renfei Wang. North-american speed limit sign detection and recognition for smart cars. In *Local Computer Networks Workshops (LCN Workshops), 2013 IEEE 38th Conference on*, pages 154–161. IEEE, 2013.
- [168] Subrata Kumar Kundu and Patrick Mackens. Speed limit sign recognition using mser and artificial neural networks. In *Intelligent Transportation Systems (ITSC), 2015 IEEE 18th International Conference on*, pages 1849–1854. IEEE, 2015.

- [169] Aryuanto Soetedjo and I Komang Somawirata. Speed limit traffic sign classification using multiple features matching. In *International Conference on Information Theoretic Security*, pages 210–217. Springer, 2017.
- [170] Chi-Yi Tsai, Hsien-Chen Liao, and Yen-Chang Feng. A novel translation, rotation, and scale-invariant shape description method for real-time speed-limit sign recognition. In *Advanced Materials for Science and Engineering (ICAMSE), International Conference on*, pages 486–488. IEEE, 2016.
- [171] Samuel L Gomes, Elizângela de S Rebouças, Edson Cavalcanti Neto, João P Papa, Victor HC de Albuquerque, Pedro P Rebouças Filho, and João Manuel RS Tavares. Embedded real-time speed limit sign recognition using image processing and machine learning techniques. *Neural Computing and Applications*, pages 1–12, 2016.
- [172] Bing Liu, Huaping Liu, Xiong Luo, and Fuchun Sun. Speed limit sign recognition using log-polar mapping and visual codebook. *Advances in Neural Networks–ISNN 2012*, pages 247–256, 2012.
- [173] Kwangyong Lim, Taewoo Lee, Changmok Shin, Soonwook Chung, Yeongwoo Choi, and Hyeran Byun. Real-time illumination-invariant speed-limit sign recognition based on a modified census transform and support vector machines. In *Proceedings of the 8th International Conference on Ubiquitous Information Management and Communication*, page 92. ACM, 2014.
- [174] Fabien Moutarde, Alexandre Bargeton, Anne Herbin, and Lowik Chanussot. Robust on-vehicle real-time visual detection of american and european speed limit signs, with a modular traffic signs recognition system. In *Intelligent Vehicles Symposium, 2007 IEEE*, pages 1122–1126. IEEE, 2007.
- [175] Khairul Anuar Ishak, M Mohd Sani, and Nooritawati Md Tahir. A speed limit sign recognition system using artificial neural network. In *Research and Development, 2006. SCOReD 2006. 4th Student Conference on*, pages 127–131. IEEE, 2006.
- [176] Marcin L Eichner and Toby P Breckon. Integrated speed limit detection and recognition from real-time video. In *Intelligent Vehicles Symposium, 2008 IEEE*, pages 626–631. IEEE, 2008.
- [177] Shigeharu Miyata. Automatic recognition of speed limits on speed-limit signs by using machine learning. *Journal of Imaging*, 3(3) :25, 2017.
- [178] Gang Yan, Ming Yu, Shuo Shi, and Chao Feng. The recognition of traffic speed limit sign in hazy weather. *Journal of Intelligent & Fuzzy Systems*, 33(2) :873–883, 2017.
- [179] Maurice Peemen, Bart Mesman, and Henk Corporaal. Speed sign detection and recognition by convolutional neural networks. In *Proceedings of the 8th International Automotive Congress*, pages 162–170, 2011.
- [180] Yuan Li, Andreas Møgelmoose, and Mohan Manubhai Trivedi. Pushing the “speed limit” : High-accuracy us traffic sign recognition with convolutional neural networks. *IEEE Transactions on Intelligent Vehicles*, 1(2) :167–176, 2016.

## Conclusion générale

---

- [181] Radu Timofte, Karel Zimmermann, and Luc Van Gool. Multi-view traffic sign detection, recognition, and 3d localisation. *Machine vision and applications*, 25(3) :633–647, 2014.
- [182] Rachid Belaroussi, Philippe Foucher, Jean-Philippe Tarel, Bahman Soheilian, Pierre Charbonnier, and Nicolas Papanicolaou. Road sign detection in images : A case study. In *Pattern Recognition (ICPR), 2010 20th International Conference on*, pages 484–488. IEEE, 2010.
- [183] Andreas Møgelmo, Mohan M Trivedi, and Thomas B Moeslund. Learning to detect traffic signs : Comparative evaluation of synthetic and real-world datasets. In *Pattern Recognition (ICPR), 2012 21st International Conference on*, pages 3452–3455. IEEE, 2012.
- [184] Cosmin Grigorescu and Nicolai Petkov. Distance sets for shape filters and shape recognition. *IEEE transactions on image processing*, 12(10) :1274–1286, 2003.
- [185] Andreas Møgelmo, Mohan M Trivedi, and Thomas B Moeslund. Vision-based traffic sign detection and analysis for intelligent driver assistance systems : Perspectives and survey. *IEEE Trans. Intelligent Transportation Systems*, 13(4) :1484–1497, 2012.