RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

---

UNIVERSITE ECHAHID HAMMA LAKHDAR - EL OUED

FACULTÉ DES SCIENCES EXACTES

Département D'Informatique

Mémoire de Fin D'étude

Présenté pour l'obtention du Diplôme de

MASTER ACADEMIQUE

Domaine : **Mathématique et Informatique**

Filière : **Informatique**

Spécialité : **Systèmes Distribués et Intelligence Artificielle**

Présenté par :

- SEBOUAI HICHAM
- DEGHBAR ASMA

Thème

Détection Et Reconnaissance Des Panneaux De Signalisation Basé Sur L'apprentissage Profond

Soutenue le xx-xx- 2023 Devant le jury :

M. **Laouid Abdelkader** MCA Encadreur

M. **Guia Sana Sahar** MAA Encadreur

M. …… MCA Président

M. …… MAA Rapporteur

Année Universitaire : 2022/2023

REMERCIEMENT

Avant tout, nous voudrions exprimer notre gratitude à Dieu qui nous a donné la force, la volonté, le courage et la patience nécessaires pour entreprendre cet humble acte. Nous tenons également à remercier nos encadrants, Proffessure Guia Sana Sahar et le professeur Laouid Abdelkader, pour avoir guidé notre travail et nous avoir prodigué des conseils pendant notre séjour.

Nous leur sommes reconnaissants de leur confiance et de leur aide précieuse et nous leur exprimons nos sincères remerciements.

Nous remercions également les membres du jury qui ont accepté de juger notre humble travail. Nous remercions sincèrement tous les professeurs et administrateurs du Département d'informatique, en particulier ceux qui nous ont enseigné pendant nos études. Nous tenons à exprimer notre gratitude à tous ceux qui ont contribué de près ou de loin à l'élaboration de ce travail.

RÉSUMÉ :

La détection et la reconnaissance des panneaux de signalisation sont depuis longtemps un domaine d'étude qui intéresse les chercheurs. Pour résoudre ce problème de nombreuses solutions utilisant différentes techniques ont été proposées. L'une de ces solutions développées récemment est l'apprentissage en profondeur. (Deep Learning), qui est un type d'intelligence artificielle dérivée de l'apprentissage automatique, où la machine est capable d'apprendre par lui-même, et a obtenu d'excellents résultats et une grande précision dans diverses applications de vision par ordinateur.

Dans ce travail, nous avons effectué la détection des feux de circulation dans les images de scène naturelles sur l'ensemble de données sdv data set. Nous avons obtenu cet ensemble de données en traitant les images originales du dataset, ce qui nous a permis de réduire leur taille tout en conservant leurs caractéristiques pertinentes. Ensuite nous avons utilisé le modèle YOLO v5 pour détecter les panneaux de signalisation dans ces images. Les résultats obtenus étaient très bons par rapport à l'ensemble des données d'origine, bien que nous ayons dû réduire la taille de l'ensemble de données de 50 000 images à 10 000 images par manque de ressources.

Mots clés: Détection d'objet, reconnaissance et détection de panneaux de signalisation, deep Learning, conduite autonome, YOLO v5,SDV.

ABSTRACT :

The literature review has long studied the detection and recognition of traffic signs. To this end, Researchers proposed numerous solutions using different techniques. One of the recently developed solutions is the deep learning technique, a type of artificial intelligence derived from machine learning, where the machine is capable of learning independently and has achieved excellent results and high accuracy in various computer vision tasks.

We detected traffic signs in natural scene images using the SDV dataset obtained by processing the original images of the dataset, which allowed us to reduce their size while preserving their relevant characteristics. Then, we used the YOLO v5 model to detect traffic signs in these images. The results were perfect compared to the original dataset, although we had to reduce the dataset size from 50,000 to 10,000 images due to resource constraints.

Keywords: Object detection, detection and recognition of traffic signs, deep Learning, autonomous driving, YOLO v5,SDV.

ملخص:

يعد إكتشاف اشارات المرور و التعرف عليها مجالا للدراسة طالما أثار إهتمام الباحثين.حيث أن لمشكلة الكشف على اشارات المرور و التعرف عليها العديد من الحلول بإستخدام تقنيات مختلفة.أحد الحلول التي تم إستخدامها و تطويرها مؤخرا هو التعلم العميق (Deep Learning)و هو نوع من الذكاء الإصطناعي مشتق من التعلم الآلي(Machine Learning)حيث تكون الآلة قادرة على التعلم بنفسها و قد حققت نتائج ممتازة و دقة عالية في مختلف تطبيقات الرؤية الحاسوبية .

في هذا العمل، أجرينا إكتشاف اشارات المرور في صور المشاهد الطبيعية على (sdv data set) والتي نحصل عليها بمعالجة صور الأصلية لـ data setاي (RGB original image) مما يؤدي إلى تقليل حجمها مع المحافظة على خصائصها. بعد ذلك قمنا باستخدام خوارزمية معالجة الصور (yolo v5) لإكتشاف اشارات المرور و التعرف عليها حيث كانت نتائج (sdv data set) جيدة جدا مقارنة بـ (original dataset) رغم أننا اضطررننا إلى تقليل حجم (data set)من 50000صورة إلى 10000صورة بسبب قلة الموارد.

الكلمات المفتاحية:الكشف عن الأشياء, التعرف على اشارات المرور واكتشافها, التعلم العميق, القيادة الذاتية .SDV,YOLO v5

TABLE DES MATIÈRES

TABLE DES FIGURES

LISTE DES TABLEAUX

INTRODUCTION GÉNÉRALE

Selon les Nations Unies, les décès sur les routes devraient augmenter de 50 entre 2010 et 2020, Cela a conduit à la création de la première Décennie d'action pour la sécurité routière 2011 a inversé cette tendance. Les systèmes d'aide à la conduite peuvent faire la différence Il est important de réduire le nombre d'accidents en automatisant certaines tâches telles que Détecter la fatigue, les piétons et reconnaître les panneaux de signalisation. progrès La technologie dans le domaine automobile vise à développer des systèmes pour aider Des conducteurs encore plus intelligents pour minimiser les erreurs de conduite et les accidents. plus important De plus en plus de personnes conduisent désormais avec des assistants d'aide à la conduite Moins de stress. les panneaux de signalisation peuvent être informatifs Crucial pour les conducteurs mais souvent négligé pour les raisons suivantes la fatigue ou des conditions météorologiques défavorables. La reconnaissance automatique des panneaux routiers est considérée comme une fonctionnalité essentielle des véhicules intelligents afin de réduire le nombre d'accidents sur les routes. Les systèmes d'assistance à la conduite peuvent aider à détecter et à identifier les panneaux routiers, permettant ainsi aux conducteurs de prendre des décisions plus éclairées et de garantir leur sécurité sur la route. La reconnaissance automatique des panneaux routiers nécessite trois étapes distinctes, à savoir la localisation, la détection et la reconnaissance. Pour la détection, un algorithme de segmentation des couleurs est souvent utilisé, impliquant la conversion des images en divers espaces colorimétriques tels que RGB, HSV, HSI, CIElab et YCbCr. Bien que l'espace RGB [1] [2] [3] [4] soit couramment utilisé, il peut être sensible aux fluctuations de la lumière. Par conséquent, l'espace HSV est souvent privilégié car la teinte H est censée être invariante aux variations de luminance. Le seuillage est souvent utilisé comme méthode de segmentation,[5] mais certains auteurs préfèrent utiliser des filtres ou une agrégation dynamique de pixels pour contourner les problèmes de choix de seuil. Cependant, les méthodes

les plus efficaces peuvent entraîner des coûts en termes de Temps de calcul élevé. Ce passage décrit les différentes manières de détecter les panneaux les routes, y compris les méthodes basées sur la couleur les plus courantes, et Méthodes basées sur la forme, moins utilisées. Parmi elles, les informations Les couleurs sont soit prétraitées, soit ignorées. Plusieurs algorithmes ont été développés pour Détection de points de repère, comme la correspondance de modèles, la transformation de Hough et transformer la symétrie radiale (TSR) [6]. D'autres auteurs ont également choisi Utilisez la transformation [7] HOG (Histogram of Oriented Gradients).

Les méthodes traditionnelles de détection des panneaux routiers qui se basent sur la couleur et la géométrie sont souvent limitées en raison de leur manque de robustesse face aux variations d'éclairage, d'échelle, d'occlusions et de rotations. Afin de résoudre ce problème, des méthodes d'apprentissage ont été développées, telles que l'algorithme AdaBoost qui utilise des ondelettes de Haar et qui a été proposé par Viola et Jones, ainsi que l'utilisation combinée de l'Adaboost et du SVM par Chen et Priscariu. Des approches alternatives, telles que les algorithmes génétiques, les réseaux de neurones et les CNN sont également utilisées pour la détection des panneaux routiers. Les méthodes basées sur les CNN sont particulièrement précises, mais elles sont également plus complexes et nécessitent un temps de formation plus long.

La phase de reconnaissance des panneaux routiers peut être effectuée à l'aide de méthodes d'apprentissage classiques ou de deep learning. Les descripteurs tels que HOG, LBP, SIFT ou SURF sont souvent utilisés dans les méthodes classiques, mais leur précision dépend de leur capacité à représenter les caractéristiques distinctives. Pour améliorer la précision, certains chercheurs se tournent vers les méthodes de deep learning, mais cela nécessite une grande quantité de données annotées et un temps d'apprentissage considérable.[8] [9] [10] [11]

La performance de la détection et de la reconnaissance des panneaux de signalisation dans le monde réel peut être affectée par différents facteurs tels que les variations d'éclairage, les changements de conditions météorologiques, les occlusions causées par d'autres objets, les rotations des panneaux capturés lors de virages et les changements de taille des panneaux. Tous ces éléments peuvent potentiellement réduire l'efficacité du système de reconnaissance des signes[12] [13].

• **définition de problème :** Pour détecter et identifier les feux de circulation, nous sommes confrontés à de nombreux problèmes, notamment : Changé en miettes en raison du soleil et de la pluie, ce qui le rend difficile à reconnaître,Conditions météorologiques, La présence de nuances ou de couleurs similaires aux feux de circulation, Faible qualité de l'image, Le mouvement ou la vibration du véhicule rend l'image capturée par la caméra floue et difficile à reconnaître, Variété d'angles de prise de la photo.

• **objectifs de la recherche :** Traiter la distorsion de réflexion, le flou de mouvement et les erreurs de capture vidéo causées par la gravure ou le mouvement de la caméra, Reconnaître les feux de circulation et garder leur emplacement, Aider les conducteurs à réduire les accidents de la circulation.

• **structure de la thèse :**

Chapitre 1 : Fournit une introduction générale au système avancé d'aide à la conduite et à son concept, ce chapitre décrit l'importance de ce système et de ses applications en mentionnant les caractéristiques des feux de circulation.

Chapitre 2 : Ce chapitre décrit une introduction générale à la détection d'objets, son concept et ses avantages. Il existe différentes approches de la détection d'objets, y compris les approches classiques (SIFT, HOG, Framework de détection d'objets Viola–Jones object) et les méthodes basées sur l'apprentissage profond. (R-CNN, Fast R-CNN, Faster R-CNN, cascade R-CNN, SSD, YOLO, Single-Shot Refinement Neural, Retina-Net).

Chapitre 3 : Dans ce chapitre, nous introduisons le modèle d'apprentissage en profondeur et yolo modal utilisé dans le projet, ainsi qu'une explication de l'ensemble de données utilisé.

Chapitre 4 : Les différentes étapes de code et les résultats obtenus sont donnés Nous terminons ensuite par la conclusion générale.

CHAPITRE 1

SYSTÈME AVANCÉ D'AIDE À LA CONDUITE (EN ANGLAIS ADAS ADVANCED DRIVING ASSISTANCE SYSTEM)

1.1 Introduction :

ADAS (Advanced Driver Assistance System) est un dispositif technique Conçu pour aider les conducteurs à rester en sécurité pendant la conduite. qui comprend de nombreux Une fonction réactive qui peut éviter les collisions [14]. Le premier système ADAS Cela inclut des technologies telles que l'avertissement de sortie de voie et le régulateur de vitesse adaptatif. Des fonctionnalités supplémentaires ont été ajoutées au fil du temps. Au cours des dernières années, avec l'avènement de la voiture autonome, les systèmes ADAS se sont encore améliorés, offrant des fonctionnalités telles que l'automatisation de la conduite sur autoroute, la conduite autonome à basse vitesse et la conduite autonome en ville.

En général, les systèmes ADAS ont joué un rôle clé dans l'amélioration de la sécurité routière et continuent de se développer à mesure que de nouvelles technologies apparaissent. Il est probable que ces systèmes continueront à évoluer dans les années à venir pour offrir une aide encore plus précieuse aux conducteurs sur la route.

1.2 Importance de système avancé d'aide à la conduite

— d'éviter la survenue de situations dangereuses pouvant aboutir à un accident.

— de libérer le conducteur des tâches susceptibles d'altérer sa vigilance.

— d'assister le conducteur pour la perception de l'environnement.

— permettre à l'automobile de percevoir les risques et de réagir en conséquence.

L'atteinte de ces objectifs est rendue possible grâce à l'accomplissement des missions spécifiques assignées aux technologies que regroupent les systèmes ADAS.[15]

1.3 Niveaux d'aide à la conduite :

Niveau 0 (Pas d'automatisation) : Aucune aide à la conduite. Le conducteur est entièrement responsable de toutes les tâches de conduite.[16]

Niveau 1(Conduite assistée "eyes on-hands on") : Aide à la conduite de niveau 1. Le véhicule est équipé de fonctionnalités telles que le régulateur de vitesse adaptatif, l'aide au maintien de voie, la détection de collision, etc. Cependant, le conducteur est toujours responsable de la conduite.[16]

Niveau 2(Automatisation partielle "eyes on-hands off") :

Aide à la conduite de niveau 2. Le véhicule peut prendre en charge certaines tâches de conduite,

telles que la direction et la vitesse, mais le conducteur doit toujours surveiller l'environnement de conduite et être prêt à reprendre le contrôle du véhicule à tout moment.[16]

Niveau 3(Automatisation sous conditions "eyes off-hands off") :
A ce niveau, le conducteur ne conduit plus le véhicule tant qu'il dispose des fonctions automatiques Tick est Engagement actif, même s'il reste derrière le volant. Cependant, le conducteur doit rester dans Soyez toujours vigilant et supervisez la conduite afin de pouvoir reprendre le contrôle Immédiatement lorsque le système enseigne le véhicule. L'automatisation de niveau 3 garantit Le véhicule de contrôle ne peut pas être démarré dans des conditions restreintes à moins que le Toutes les conditions préalables sont remplies. Par exemple, pour fournir une aide à la conduite dans les embouteillages Scutching (Traffic Jam Assist) ou Automatic Lane Keeping System (ALKS), par exemple Pilot tesla automatique.[16]

Niveau 4(Conduite autonome sous conditions "eyes off-hands off-mind off") :
À ce niveau, le conducteur n'a plus forcément besoin de reprendre la conduite, le véhicule étant en mesure de s'arrêter seul dans un endroit sûr si nécessaire, sans avoir recours à l'humain. Si la supervision de la conduite n'est plus obligatoire, un volant reste cependant présent pour permettre un contrôle manuel du véhicule. L'automatisation de niveau 4 reste cependant soumise à des conditions d'utilisations strictes et toutes doivent être remplies, comme au niveau 3. Il s'agit là d'un véhicule autonome, mais qui n'est pas apte à faire face à toutes les conditions.[16]

Niveau 5(Véhicule autonome "driverless") :
Ultime stade de l'automatisation, le niveau 5 correspond aux véhicules totalement autonomes. Le poste de pilotage ne comprend plus de volant et l'intégralité de la conduite est gérée par le système. Celui-ci est en mesure de faire face à toutes les situations, toutes les routes et dans toutes les conditions sans la moindre supervision.[16]

1.4 système de conduite automatisé :

Le système de conduite automatisé est une technologie qui permet à une voiture de se conduire elle-même, sans intervention humaine. Les véhicules équipés de cette technologie peuvent utiliser des capteurs, des caméras, des radars et des cartes pour percevoir leur environnement et déterminer la meilleure voie à suivre. Ils peuvent également utiliser des algorithmes de traitement de la conduite pour contrôler la vitesse, la direction et les fonctions de freinage. Il existe différents niveaux de conduite automatisée, allant de l'assistance à la conduite (comme le régulateur de vitesse adaptatif) à la conduite complètement autonome.[17]

1.5 Application d'ADAS :

Il existe plusieurs types d'applications avancées pour les systèmes d'assistance à la conduite, notamment : [18]

1.5.1 Reconnaissance de panneaux de signalisation routière :

Les systèmes avancés d'aide à la conduite (ADAS) sont des technologies de sécurité avancées qui peuvent aider les conducteurs à éviter les collisions et à se conformer aux règles de la route. Les applications ADAS pour la reconnaissance des panneaux routiers font partie de ces systèmes et utilisent l'intelligence artificielle pour reconnaître et identifier les différents panneaux de signalisation routière. Les applications ADAS pour la reconnaissance des panneaux routiers peuvent utiliser des caméras et des capteurs intégrés dans le véhicule pour analyser l'environnement routier. Les images capturées sont ensuite analysées par des algorithmes de vision par ordinateur, qui peuvent identifier les panneaux de signalisation routière et afficher des informations pertinentes pour le conducteur sur le tableau de bord ou sur l'écran d'affichage tête haute.

Ces applications peuvent fournir des informations telles que la vitesse autorisée sur une route, les restrictions de dépassement, les emplacements de radars et les conditions de circulation. Certaines applications ADAS pour la reconnaissance des panneaux routiers peuvent également émettre des alertes visuelles ou sonores si le conducteur ne se conforme pas aux règles de la route.

Cependant, il est important de noter que ces applications ne sont pas toujours précises à 100, et les conducteurs doivent toujours être vigilants et se conformer aux panneaux de signalisation réels plutôt que de se fier uniquement à l'application ADAS.[18]

1.5.2 Reconnaissance de feu de circulation :

Les systèmes avancés d'aide à la conduite (ADAS) comprennent également des applications pour la reconnaissance des feux de circulation. Ces applications utilisent des capteurs et des caméras intégrés dans le véhicule pour identifier et reconnaître les feux de signalisation routière, notamment les feux de circulation.

Les applications ADAS pour la reconnaissance des feux de circulation peuvent utiliser des algorithmes de vision par ordinateur pour analyser les images capturées par les caméras du véhicule. Les informations sur l'état des feux de circulation peuvent ensuite être affichées sur

le tableau de bord du véhicule, indiquant au conducteur si le feu est vert, rouge ou orange. Certains systèmes ADAS pour la reconnaissance des feux de circulation peuvent également fournir des alertes sonores ou visuelles pour avertir le conducteur en cas de feu rouge ou de changement de feu. Il est important de noter que ces applications ne sont pas parfaites à 100 et qu'il est toujours essentiel pour les conducteurs de se conformer aux règles de la route et de rester attentifs à leur environnement. Les conducteurs doivent toujours être prêts à réagir en cas de changement de situation de conduite, indépendamment des informations fournies par l'application ADAS pour la reconnaissance des feux de circulation.[18]

1.6 les panneaux de signalisation :

Les panneaux de signalisation sont des dispositifs routiers utilisés pour transmettre des informations, des avertissements et des réglementations aux usagers de la route. Ils sont généralement placés le long des routes et des voies de circulation pour guider les conducteurs, les piétons et les cyclistes et assurer la sécurité routière.Les panneaux de signalisation sont conçus de manière à être facilement visibles et compréhensibles. Ils utilisent des symboles, des pictogrammes et des mots courts pour communiquer rapidement et efficacement les messages aux usagers de la route. Les panneaux de signalisation peuvent indiquer des informations telles que les limitations de vitesse, les directions, les destinations, les zones de danger, les intersections, les passages pour piétons, les interdictions, les obligations, les conseils de sécurité, etc.[19]

1.6.1 Principe :

Les panneaux de signalisation sont des éléments importants de la signalisation routière et ils sont utilisés pour communiquer des informations, des avertissements et des réglementations aux conducteurs sur les routes. Voici quelques principes fondamentaux des panneaux de signalisation :[20]

• **Uniformité :** Les panneaux de signalisation suivent des normes et des conventions spécifiques pour assurer une uniformité nationale et internationale. Cela facilite la reconnaissance et la compréhension des panneaux par les conducteurs, quel que soit l'endroit où ils se trouvent.

• **Visibilité :**Les panneaux de signalisation sont conçus pour être visibles et reconnaissables à distance, de jour comme de nuit. Ils utilisent des couleurs vives, des symboles simples et des polices de caractères lisibles pour maximiser leur lisibilité.

• **Symboles et pictogrammes :** Les panneaux de signalisation utilisent souvent des sym-

boles et des pictogrammes universels pour transmettre des informations. Cela permet une compréhension rapide et facile des messages, indépendamment de la langue ou de la culture des conducteurs.

• **Clarté du message :**Les panneaux de signalisation sont conçus pour être concis et clairs. Les messages sont généralement courts et directs, utilisant des mots ou des phrases simples pour communiquer efficacement l'information nécessaire aux conducteurs.

• **Hiérarchie de l'information :** Dans les zones où plusieurs panneaux de signalisation sont présents, il y a une hiérarchie de l'information pour aider les conducteurs à lire et à comprendre les panneaux dans l'ordre approprié. Les panneaux de réglementation (par exemple, les limitations de vitesse) sont généralement placés en premier, suivis des panneaux d'avertissement (par exemple, les virages dangereux) et des panneaux d'information (par exemple, les directions).

• **Placement stratégique :** Les panneaux de signalisation sont placés de manière stratégique le long des routes pour assurer une visibilité adéquate et une compréhension claire des messages. Ils sont généralement positionnés avant les emplacements pertinents, permettant aux conducteurs d'ajuster leur conduite en conséquence.

1.6.2 Types de feux de circulation :

les feux de circulation varie selon les pays, mais généralement elle suit les étapes suivantes :[21]

1-Rouge :

Un feu rouge allumé indique que tous les usagers de la route doivent s'arrêter derrière la ligne. Le signal s'applique à tout le monde, même aux cyclistes. Ignorer le feu de circulation est un gros risque et pourrait vous causer pas mal d'ennuis..

2-Vert :

Lorsque le feu vert est allumé, cela signifie que vous pouvez continuer, tant que le chemin est libre.

3-Le feu jaune :

signifie que le feu va bientôt passer au rouge et que les conducteurs doivent être prêts à s'arrêter. .

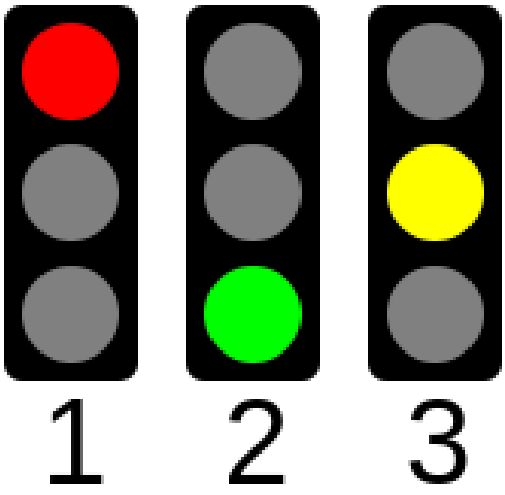

FIGURE 1.1 – séquence de feu de circulation.

1.6.3 Types de panneaux de signalisation :

1-Les panneaux de danger :

Dans les panneaux du code de la route, les panneaux de signalisation de danger sont ceux de forme triangulaire, au fond blanc et de bordure rouge. Chaque panneau annonce un danger précis à l'aide d'un symbole noir, en fonction du risque présent sur la route.[20]

FIGURE 1.2 – Les panneaux de danger.

2-Les panneaux de prescription (interdiction et obligation) :

Les panneaux de prescription, ou d'interdiction, sont de forme ronde Le fond du panneau est bleu. Lorsque vous voyez un panneau de ce genre sur la route, vous êtes obligé d'obéir à son indication. Le code de la route définit ce panneau.[20]

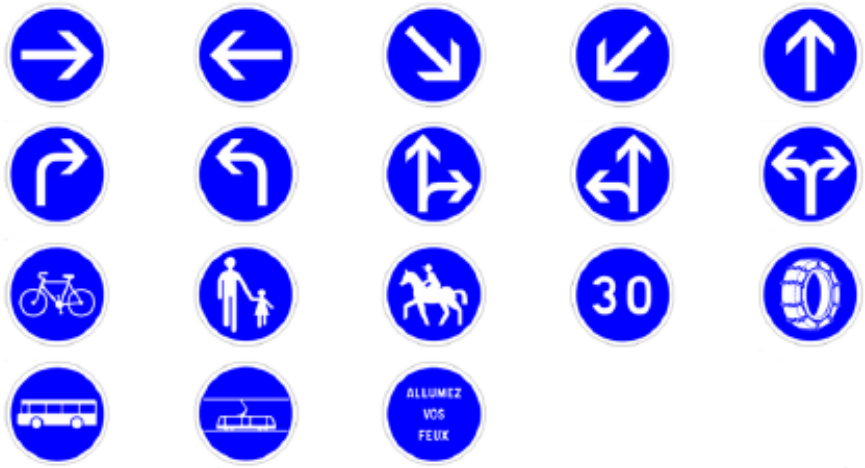

FIGURE 1.3 – Les panneaux d'obligation..

3-Les panneaux routiers d'indication :

Les panneaux de signalisation routière de type "interdiction" sont utilisés pour indiquer aux conducteurs qu'ils doivent respecter une interdiction spécifique sur la route. Ces panneaux sont généralement de forme circulaire et ont un fond blanc, jaune ou bleu, bordé d'une couronne rouge, comme illustré dans la figure. [20]

FIGURE 1.4 – Les panneaux routiers d'indication.

4-Les panneaux d'intersection et de priorité :

Les panneaux d'intersection vous préviennent du type de carrefour à l'approche et vous indiquent les règles de priorité à suivre : carrefour à sens giratoire, priorité à droite, stop, etc.[20]

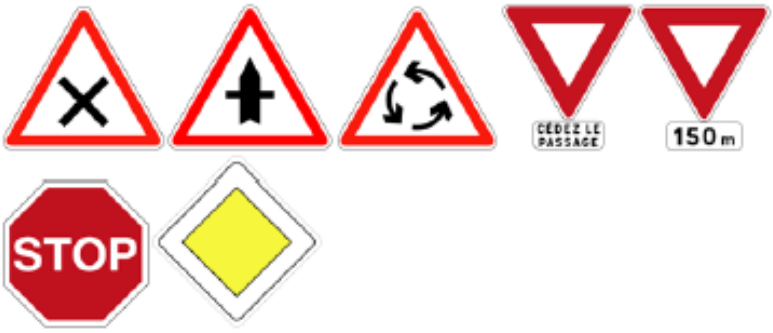

FIGURE 1.5 – Les panneaux d'intersection et de priorité.

5-Les panneaux de prescriptions particulières :

Les panneaux de signalisation routière de type "prescription particulière" peuvent être de forme carrée ou circulaire. Ils ont un fond bleu avec un symbole ou une inscription de couleur claire, ou un fond clair avec un symbole ou une inscription de couleur foncée.[20]

FIGURE 1.6 – Les panneaux de prescriptions particulières.

1.7 Conclusion :

Dans ce chapitre, nous discutons de l'histoire et du concept des systèmes avancés d'aide à la conduite. Nous avons expliqué son importance et examiné les différents niveaux d'automatisation de la conduite. Nous décrivons également certaines applications des systèmes avancés d'aide à la conduite dans la reconnaissance des panneaux routiers et des panneaux de signalisation. Enfin, nous avons fourni un aperçu des différents types de feux de circulation et de signalisation routière.

CHAPITRE 2

DÉTECTION D'OBJET (OBJECT DETECTION)

2.1 Introduction :

La détection d'objet est une tâche importante en vision par ordinateur qui consiste à détecter la présence d'objets dans une image ou une vidéo et à localiser leur position en dessinant une boîte englobante autour de l'objet détecté. Cette tâche est utilisée dans de nombreuses applications telles que la reconnaissance de visage, la détection de véhicules, la surveillance de sécurité, la navigation autonome, etc.

La détection d'objet est un problème complexe car les objets peuvent varier considérablement en termes de taille, de forme, de couleur, de texture et de position dans l'image. En outre, les objets peuvent être partiellement occultés, superposés ou présentés sous des angles différents, ce qui complique la tâche de détection. Pour résoudre ce problème, de nombreuses méthodes ont été développées pour extraire des caractéristiques pertinentes des images et pour entraîner des modèles d'apprentissage automatique pour détecter et localiser les objets. Au fil des années, les méthodes de détection d'objet ont évolué de manière significative, passant des méthodes basées sur les caractéristiques à des méthodes basées sur les réseaux de neurones convolutifs (CNN) profonds. Les méthodes basées sur les CNN ont révolutionné la détection d'objet en permettant l'apprentissage de caractéristiques à partir de données d'entraînement, ce qui permet d'obtenir des taux de précision élevés, même dans des situations difficiles.

2.2 Intérêt :

La détection d'objet est une tâche cruciale pour de nombreuses applications telles que la conduite autonome, la vidéosurveillance, la reconnaissance de visage, la détection de défauts dans la fabrication industrielle, etc. Voici quelques-uns des avantages de cette technologie :

— Automatisation des touches : Détection d'objets permet d'automatiser certaines tâches répétitives et fastidieuses, telles que la vidéosurveillance à temps réel ou la détection de défauts dans la fabrication industrielle. Si vous faites attention aux ingrédients, améliorez la qualité et augmentez le produit.

— Amélioration de la sécurité : la détection d'un objet peut aider à éviter la sécurité face à des dangers identifiables, tels que des armes, des explosifs, des produits chimiques, etc. Elle peut également être utilisée pour détecter les obstacles sur la route lors de la conduite autonome ou pour surveiller les zones à risque.

— Optimisation du processus : La détection d'objet peut aider à optimiser les processus en permettant une analyse automatisée de grandes quantités de données. Elle peut également

être utilisée pour identifier les tendances et les modèles dans les données, ce qui peut aider à prendre des décisions plus éclairées.

la détection d'objet est une technologie importante pour de nombreuses applications et peut offrir des avantages tels que l'automatisation des touches, l'amélioration de la sécurité, l'optimisation des processus et l'amélioration de l'expérience utilisateur.

2.3 Concept :

La détection d'objet est une tâche clé en vision par ordinateur et en intelligence artificielle qui consiste à identifier et à localiser des objets spécifiques dans des images ou des vidéos. Elle peut être utilisée pour une grande variété d'applications telles que la reconnaissance de visages, la détection de véhicules, la surveillance de la qualité des produits, la détection de maladies dans des images médicales, etc.

Le concept de détection d'objet implique généralement deux étapes principales : la proposition de régions candidates et la classification de ces régions. La première étape consiste à trouver des zones qui pourraient contenir un objet. La seconde étape consiste à classifier ces zones selon les catégories d'objets connues ou inconnues.

Il existe plusieurs approches pour la détection d'objet, qui ont évolué au fil du temps avec l'avancée des technologies de l'apprentissage en profondeur. Les méthodes traditionnelles étaient basées sur l'extraction de caractéristiques manuelles et l'utilisation de classificateurs tels que les SVM (Support Vector Machines) ou les Adaboost. Cependant, ces approches sont limitées en termes de précision et de vitesse.[22]

2.4 Méthodes :

2.4.1 Approches classiques (Non-neural approaches) :

Framework de detection d'objets Viola–Jones object :

Le Viola-Jones object detection framework est une méthode classique de détection d'objets dans des images. Cette technique utilise des caractéristiques en forme de rectangles pour détecter des objets dans une image. Elle a été initialement conçue pour la détection de visages, mais elle peut également être utilisée pour la détection de panneaux de signalisation routière.[23]

— Caractéristiques :

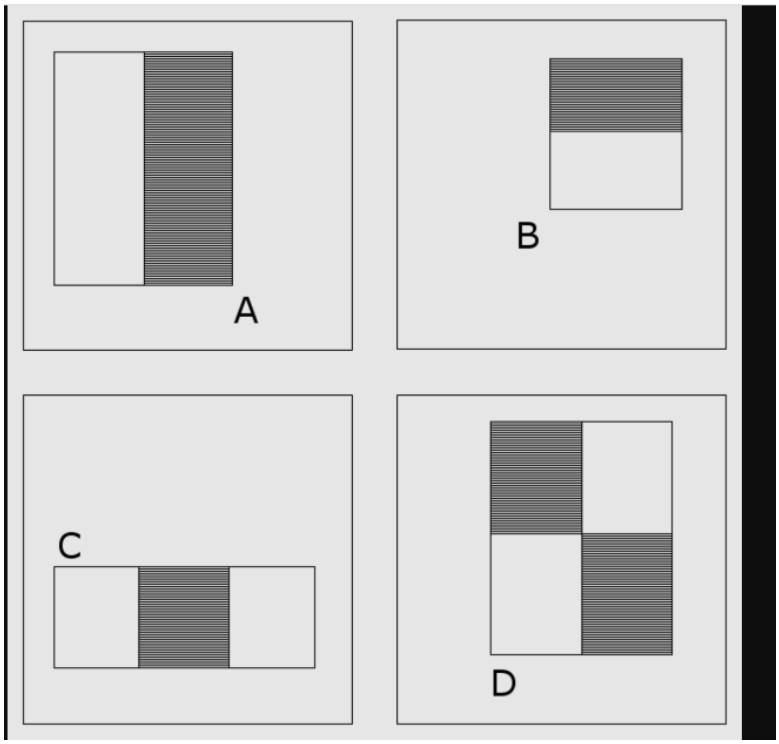

FIGURE 2.1 – types de caractéristiques utilisées par Viola et Jones.

Viola et Jones proposent d'utiliser un certain nombre de caractéristiques, plutôt que de travailler directement sur les valeurs de pixels, ce qui est à la fois plus efficace et plus rapide. Ce sont des caractéristiques quasi-Haar (en), qui sont des différences de sommes de pixels calculées dans des zones rectangulaires. La figure montre les quatre types caractéristiques proposées par Viola et Jones, dans lesquelles la somme de pixels en gris est soustraite de la somme des pixels blancs. Leur nom vient de la similarité avec les ondelettes de Haar proposées comme caractéristiques par Papageorgiou et al., et dont s'inspirent Viola et Jones.
Pour calculer efficacement ces caractéristiques sur une image, les auteurs proposent également une méthode rapide, qu'il appellent « image intégrale ». C'est une représentation sous la forme d'une image, de même taille que l'image d'origine, qui en chacun de ses points contient la somme des pixels situés en dessus et à gauche de ce point.[23]

Scale-invariant feature transform (SIFT) :

La détection des panneaux de signalisation est une tâche importante dans la conduite autonome et la sécurité routière en général. Le SIFT (Scale-Invariant Feature Transform) est une méthode de traitement d'images largement utilisée pour la détection d'objets, car elle permet de détecter des caractéristiques robustes et invariantes à l'échelle. Voici les étapes principales pour détecter les panneaux de signalisation avec SIFT [24] :

• **Collecte de données :** Il est nécessaire de collecter une base de données d'images de panneaux de signalisation pour entraîner le modèle. Ces images doivent couvrir une variété de conditions d'éclairage, d'angles de vue et de distances.[24]

• **Extraction de caractéristiques SIFT :** Pour chaque image de panneau de signalisation, extrayez des caractéristiques SIFT en utilisant des bibliothèques de traitement d'images comme OpenCV. Les caractéristiques SIFT sont des points d'intérêt dans l'image qui sont robustes à l'échelle et à la rotation.[24]

• **Construction d'un dictionnaire de visualisation :** Les caractéristiques SIFT extraites sont regroupées en un dictionnaire de visualisation en utilisant des techniques de clustering telles que k-means. Cela permet de réduire la dimensionnalité des caractéristiques et de les rendre plus facilement utilisables pour la détection d'objets.[24]

• **Entraînement d'un classificateur :** Utilisez le dictionnaire de visualisation pour entraîner un classificateur, tel qu'un SVM, pour distinguer les caractéristiques SIFT correspondant aux panneaux de signalisation de ceux qui ne le sont pas.[24]

• **Détection d'objets :** Une fois que le classificateur a été entraîné, vous pouvez utiliser la méthode de détection d'objets pour détecter les panneaux de signalisation dans de nouvelles images. Cette méthode consiste à extraire des caractéristiques SIFT à partir de l'image, puis à utiliser le classificateur pour déterminer si ces caractéristiques correspondent à celles d'un panneau de signalisation.[24]

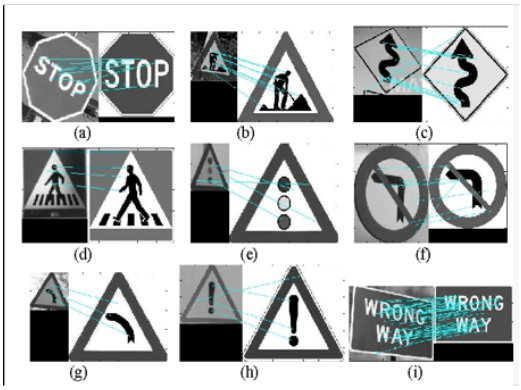

FIGURE 2.2 – reconnaissance des panneaux de signalisation routière en utilisant une version améliorée de l'algorithme SIFT..

Histogram of oriented gradients (HOG) :

La détection de reconnaissance de panneaux de signalisation routière par Histogram of oriented gradients (HOG) est une technique populaire en vision par ordinateur pour détecter

les objets dans les images. Elle est basée sur l'utilisation de l'histogramme des gradients orientés pour extraire des caractéristiques significatives de l'image. Le processus de détection de reconnaissance de panneaux de signalisation routière par HOG implique plusieurs étapes [24] :

• **Préparation de l'ensemble de données :**Un ensemble de données contenant des images de panneaux de signalisation routière est préparé. Chaque image est annotée avec l'emplacement et le type du panneau de signalisation.[24]

• **Extraction de caractéristiques :**L'histogramme des gradients orientés est calculé pour chaque image dans l'ensemble de données. Cela implique de diviser chaque image en petites régions et de calculer l'orientation du gradient pour chaque pixel dans chaque région. Les orientations des gradients sont ensuite regroupées en bins et un histogramme est calculé pour chaque région.[24]

• **Entraînement du classificateur :**Un classificateur est entraîné sur les caractéristiques extraites à partir de l'ensemble de données. Les algorithmes de classification courants incluent SVM (Support Vector Machine) et AdaBoost.[24]

• **Détection de panneaux de signalisation :** L'algorithme de détection HOG est appliqué à chaque image en utilisant le classificateur entraîné pour détecter les régions de l'image qui correspondent à des panneaux de signalisation routière. Une fois que les régions sont détectées, les panneaux de signalisation sont extraits de l'image.[24]

2.4.2 Approches basées sur l'apprentissage profond (Neural network approaches) :

Region Proposals (R-CNN, Fast R-CNN, Faster R-CNN, cascade R-CNN) :

• **R-CNN :**Pour contourner le problème de la sélection d'un grand nombre de régions, Ross Girshick et al. a proposé une méthode où nous utilisons la recherche sélective pour extraire seulement 2000 régions de l'image et il les a appelées propositions de régions. Par conséquent, maintenant, au lieu d'essayer de classer un grand nombre de régions, vous pouvez simplement travailler avec 2000 régions. Ces 2000 propositions de régions sont générées à l'aide de l'algorithme de recherche sélective décrit ci-dessous.[25]

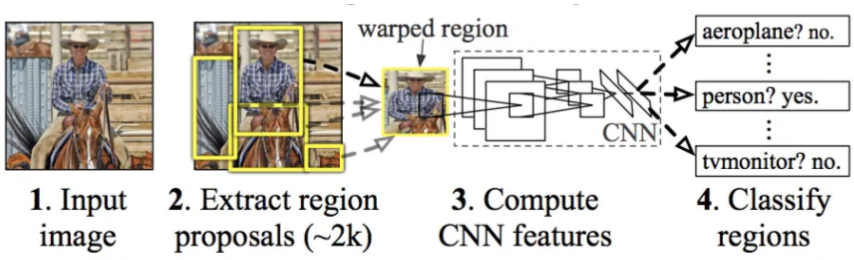

FIGURE 2.3 – R-CNN :Régions avec fonctionnalités CNN.

• **Fast R-CNN :** améliore la vitesse de R-CNN en utilisant une seule passe de convolution sur toute l'image pour extraire des caractéristiques, plutôt que d'extraire des caractéristiques pour chaque région proposée. Pour la reconnaissance de panneaux de signalisation, cela pourrait être particulièrement utile pour les images contenant plusieurs panneaux de signalisation, car l'ensemble de l'image pourrait être utilisé pour extraire des caractéristiques et localiser les panneaux.[25]

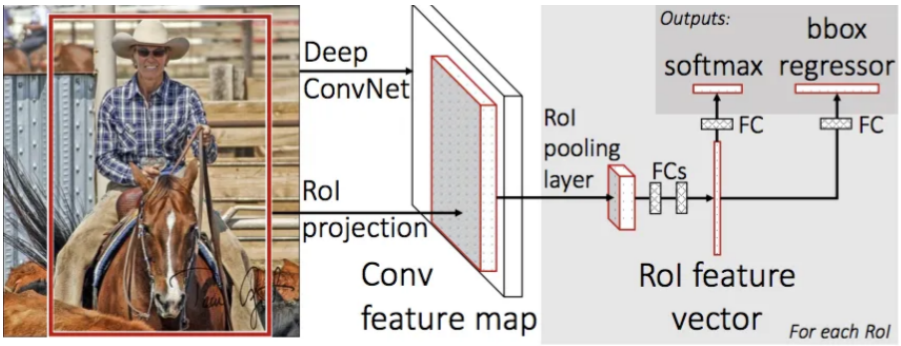

FIGURE 2.4 – Fast R-CNN.

• **Faster R-CNN :** ajoute une région de proposition de l'objet à un réseau de neurones convolutifs, appelé RPN (Region Proposal Network), qui génère automatiquement des régions proposées à partir de l'image d'entrée. Pour la reconnaissance de panneaux de signalisation, cela pourrait permettre une détection plus précise des panneaux de signalisation, car les régions proposées seraient générées de manière plus efficace et plus précise.[25]

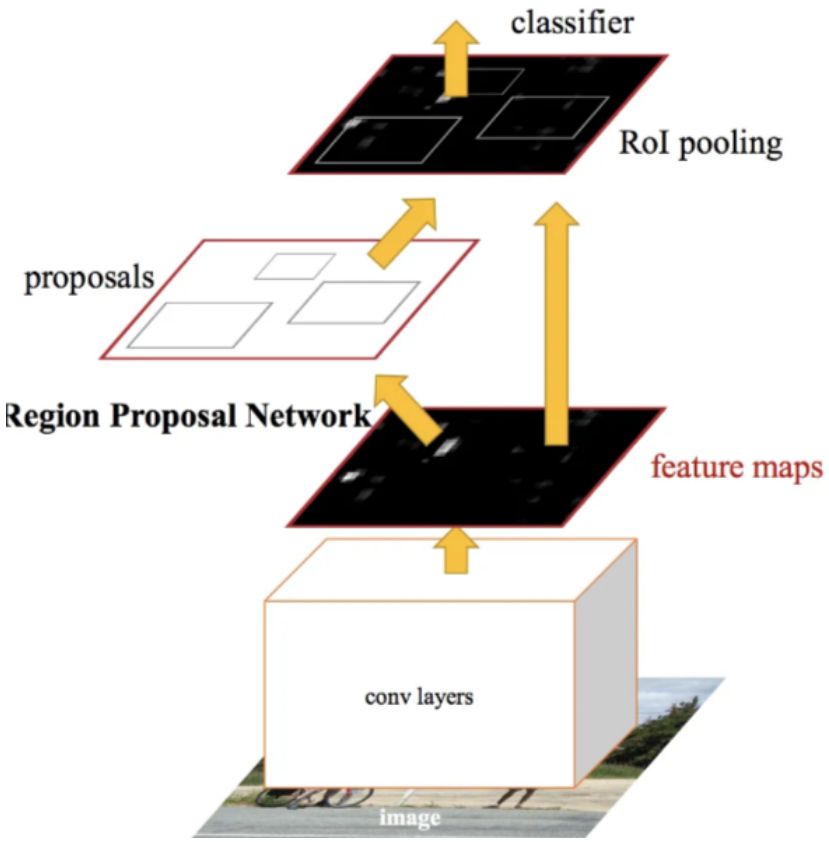

FIGURE 2.5 – Faster R-CNN.

• **Cascade R-CNN :** est une variante de Faster R-CNN qui utilise plusieurs niveaux de détection pour améliorer la précision de la détection d'objets. Pour la reconnaissance de panneaux de signalisation, cela pourrait améliorer la précision de la détection des panneaux de signalisation dans des conditions difficiles, telles que des panneaux partiellement obscurcis ou de petite taille.[25]

Single Shot MultiBox Detector (SSD) :

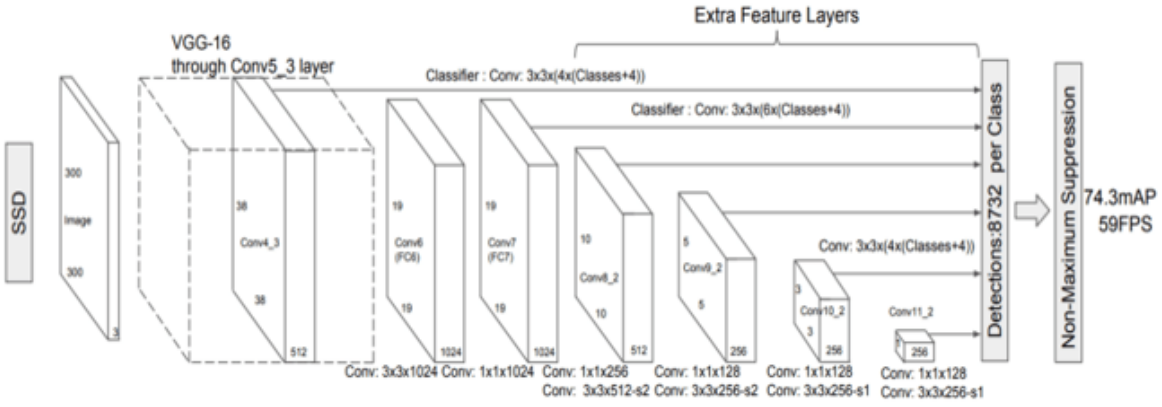

FIGURE 2.6 – Single Shot Multibox Detector.

— Collecte de données : la première étape consiste à collecter un ensemble de données d'images contenant des feux de circulation. L'ensemble de données doit inclure différents types de feux de circulation, tels que des feux de circulation rouges, jaunes et verts, avec différentes formes, tailles et arrière-plans. Les images doivent être capturées sous différents angles, distances et conditions d'éclairage pour rendre le modèle robuste à différents scénarios.[26]

FIGURE 2.7 – Localisation et reconnaissance des panneaux de signalisation sur certaines images.

— Préparation des données : l'ensemble de données est ensuite divisé en ensembles d'apprentissage, de validation et de test. Les images sont redimensionnées à une taille fixe pour les rendre compatibles avec la taille d'entrée du modèle SSD. Les images sont ensuite annotées avec des cadres de délimitation autour des feux de circulation et étiquetées avec les étiquettes de classe correspondantes.[26]

— Formation du modèle : le modèle SSD est formé sur l'ensemble de données annoté à l'aide d'un cadre d'apprentissage en profondeur. Le modèle est optimisé pour minimiser la différence entre les boîtes englobantes prévues et réelles et les étiquettes de classe. Le processus de formation consiste à ajuster les poids du modèle par rétropropagation.[26]

— Évaluation du modèle : évaluez le modèle formé sur un ensemble de données de validation distinct pour mesurer sa précision et ses performances.[26]

— Déploiement : Déployez le modèle formé pour la reconnaissance des panneaux de signalisation sur de nouvelles images ou vidéos. Cela implique de détecter les panneaux de signalisation dans l'image ou la vidéo d'entrée à l'aide de l'algorithme SSD et de dessiner des cadres de délimitation autour d'eux avec les étiquettes de classe correspondantes.[26]

Single-Shot Refinement Neural Network for Object Detection (RefineDet) :

La détection des panneaux de signalisation est une tâche importante dans la conduite autonome et les systèmes de transport intelligents. Le réseau de neurones à raffinement unique (SSR-Net) est un modèle d'apprentissage en profondeur qui peut être utilisé pour la détection d'objets, y compris la détection de panneaux de signalisation.[27]

SSR-Net est une extension de l'architecture Single Shot Detector (SSD), qui est un modèle de détection d'objet populaire. SSR-Net introduit un module de raffinement supplémentaire qui améliore encore la précision de la détection d'objets. Le module de raffinement affine la boîte englobante et la classification des objets détectés dans un processus en plusieurs étapes, conduisant à des résultats plus précis.

Pour utiliser SSR-Net pour la détection des panneaux de signalisation, le modèle doit être formé sur un ensemble de données de panneaux de signalisation. L'ensemble de données doit inclure différents types de panneaux de signalisation, de différentes formes, couleurs et tailles. L'ensemble de données doit également inclure des images de panneaux de signalisation dans diverses conditions d'éclairage et arrière-plans.

Une fois le modèle formé, il peut être utilisé pour détecter les panneaux de signalisation en temps réel. L'image d'entrée est introduite dans le modèle, et le modèle prédit l'emplacement et la classe de tous les panneaux de signalisation dans l'image. La sortie du modèle inclut les coordonnées de la zone de délimitation autour de chaque panneau de signalisation et la classe du panneau de signalisation.

L'utilisation de SSR-Net pour la détection des panneaux de signalisation présente plusieurs avantages. Premièrement, le modèle est efficace et peut fonctionner en temps réel, ce qui le rend adapté à une utilisation dans les systèmes de conduite autonome et de transport intelligent. Deuxièmement, le modèle est précis et peut détecter les panneaux de signalisation dans diverses conditions d'éclairage et arrière-plans. Troisièmement, le modèle peut détecter plusieurs panneaux de signalisation dans la même image, ce qui le rend utile pour détecter des situations de circulation complexes.

En conclusion, Single-shot Refinement Neural Network (SSR-Net) est un modèle d'apprentissage en profondeur qui peut être utilisé pour la détection des panneaux de signalisation. Avec une formation appropriée, le modèle peut détecter avec précision les panneaux de signalisation en temps réel, ce qui en fait un outil précieux pour la conduite autonome et les systèmes de transport intelligents.[27]

Retina-Net :

Retina-Net est un cadre de détection d'objets populaire qui utilise un réseau pyramidal de caractéristiques et une nouvelle fonction de perte focale pour détecter efficacement des objets à plusieurs échelles. Voici un aperçu de haut niveau de la façon dont Retina-Net peut être utilisé pour la détection des feux de circulation :[28]

- Collecte de données : collectez un grand ensemble de données d'images contenant des feux de circulation, ainsi que leurs étiquettes correspondantes (par exemple, "rouge", "vert", "jaune", "éteint", etc.). L'ensemble de données doit avoir un mélange de conditions d'éclairage, de conditions météorologiques et d'angles différents.
- Prétraitement des données : prétraitez les données en redimensionnant les images à une taille fixe, en normalisant les valeurs de pixel et en augmentant les données pour augmenter leur variabilité (par exemple, rotation, retournement, luminosité, etc.).
- Entraînement : Entraînez un modèle Retina-Net sur les données prétraitées. Le modèle doit être initialisé avec des poids pré-formés (par exemple sur ImageNet), affiné sur l'ensemble de données des feux de circulation et optimisé à l'aide de la fonction de perte focale.
- Inférence : étant donné une nouvelle image, utilisez le modèle formé pour prédire les emplacements et les étiquettes des feux de circulation. Les prédictions peuvent être post-traitées (par exemple, suppression non maximale) pour supprimer les détections en double ou qui se chevauchent.

Deformable convolutional networks :

La détection des feux de circulation est une tâche essentielle dans les systèmes de conduite autonome qui contribue au fonctionnement sûr et efficace des véhicules. Les réseaux convolutifs déformables (DCN) ont montré des résultats prometteurs dans diverses tâches de vision par ordinateur, notamment la détection d'objets, la segmentation sémantique et la segmentation d'instances. DCN peut gérer efficacement les variations géométriques et améliorer la précision de la détection d'objets.[29] Une opération de convolution déformable applique un ensemble de décalages apprenables aux emplacements d'échantillonnage de la grille régulière pour s'adapter aux transformations spatiales des objets dans l'image. Cela permet au réseau d'apprendre à détecter des objets à différentes échelles, orientations et déformations, qui sont des défis courants dans la détection des feux de signalisation.

Une étude qui applique le DCN à la détection des feux de signalisation est "Deformable Convolutional Networks for Traffic Light Detection in Autonomous Driving" . Les auteurs ont proposé un cadre basé sur DCN pour la détection des feux de circulation qui peut gérer les occlusions et les différentes tailles et positions des feux de circulation. Leur méthode a atteint des performances de pointe sur des ensembles de données de référence, y compris l'ensemble de données Bosch Small Traffic Lights et l'ensemble de données LISA Traffic Light.

Une autre étude qui utilise DCN pour la détection des feux de circulation est "Amélioration de la détection des feux de circulation à l'aide de réseaux convolutifs déformables". Les auteurs ont étendu l'approche basée sur le DCN en incorporant une connaissance préalable de l'apparence et de la forme du feu de signalisation pour améliorer la précision de la détection. Leur méthode a surpassé les approches précédentes sur l'ensemble de données Bosch Small Traffic Lights et a démontré sa robustesse face à des scénarios difficiles, tels que des conditions météorologiques défavorables et la conduite de nuit.

En résumé, les réseaux convolutifs déformables ont montré un grand potentiel dans la détection des feux de circulation et ont atteint des performances de pointe sur des ensembles de données de référence. Ces méthodes gèrent efficacement les défis de la détection d'objets dans des scénarios de conduite autonome, tels que les occlusions, les variations d'échelle et les transformations géométriques [29].

You Only Look Once (YOLO) :

Architecture "You Look Only Once" (YOLO) qui est un réseau de détection à un seul coup et a une faible latence de propagation et d'excellentes performances de détection. De nombreux réseaux de neurones modernes précis ne fonctionnent pas en temps réel et nécessitent un grand nombre de GPU pour la formation. YOLO résout ces problèmes en créant un CNN qui fonctionne en temps réel sur un GPU conventionnel, et pour lequel la formation ne nécessite qu'un seul GPU conventionnel.[30]

— Collecter et prétraiter l'ensemble de données des panneaux de signalisation : la première étape consiste à collecter un ensemble de données d'images de panneaux de signalisation et à les prétraiter. Cela peut inclure le redimensionnement des images à une taille fixe, la normalisation des valeurs de pixel et l'annotation de chaque image avec des cadres de délimitation autour des panneaux de signalisation.[30]

— Former le modèle YOLO : l'étape suivante consiste à former le modèle YOLO à l'aide de l'ensemble de données prétraité. Cela implique de définir l'architecture du modèle YOLO,

de définir des hyperparamètres et de former le modèle à l'aide de l'ensemble de données annoté.

— Testez le modèle YOLO : Une fois le modèle formé, il peut être testé sur un ensemble d'images de test pour évaluer ses performances. Cela implique d'utiliser le modèle YOLO pour détecter les panneaux de signalisation dans les images de test et de comparer les boîtes englobantes détectées avec les annotations de vérité au sol.

— Affiner le modèle YOLO : sur la base des résultats des tests, le modèle YOLO peut être affiné en ajustant les hyperparamètres, en modifiant l'architecture ou en utilisant des techniques d'augmentation de données supplémentaires.

— Déployer le modèle YOLO : Enfin, le modèle YOLO peut être déployé dans une application ou un système de détection des panneaux de signalisation en temps réel.[30]

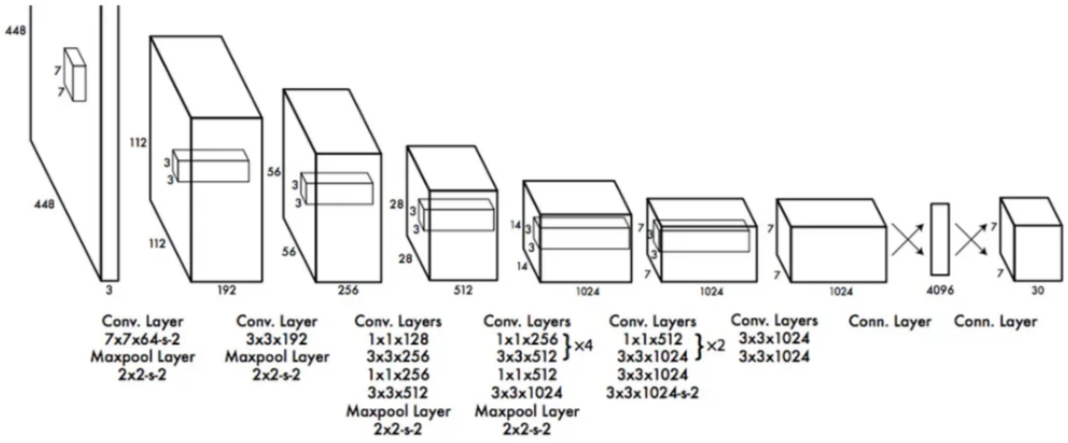

FIGURE 2.8 – Comment YOLO détecte.

2.5 Reconnaissance de panneaux de signalisation :

1. En se basant sur la propriété de saccade de la vision humaine, une équipe de chercheurs a développé un nouveau réseau de saccade en cascade qui utilise une hiérarchie de classes pour résoudre le problème de détection des panneaux de signalisation et le problème de changement de domaine. Les résultats des expériences menées sur des données de repères de panneaux de signalisation chinois (TT100K) ont montré que le détecteur proposé est comparable aux méthodes de pointe, avec une amélioration moyenne de 6 en précision et 14 en rappel pour une petite taille. De plus, le temps de détection avec un GPU est de 0,08 s par image de taille 2048 × 2048, ce qui est suffisant pour répondre aux exigences

en temps réel des applications de sécurité de conduite. En outre, le modèle proposé peut être étendu facilement pour résoudre la détection de panneaux de signalisation dans des domaines différents.[31]

2. Nous avons développé une méthode de détection de petits panneaux de signalisation qui utilise un modèle de réseau profond de bout en bout. Cette méthode est rapide et précise grâce à l'intégration de trois informations clés dans l'architecture YOLOv3 et d'autres algorithmes connexes. Nous avons également introduit une technique d'élagage pour minimiser la redondance du réseau et la taille du modèle sans compromettre la précision de détection. En outre, nous avons utilisé quatre branches de prédiction à échelle pour améliorer la précision de la prédiction multi-échelles. Nous avons ajusté la fonction de perte pour équilibrer la contribution de la source d'erreur à la perte totale. Les expériences menées sur l'ensemble de données de panneaux de signalisation Tsinghua-Tencent 100K ont montré que notre méthode est plus précise que le modèle original YOLOv3 et plus rapide que les autres méthodes de la littérature, avec une amélioration de la vitesse de détection de 1,9 à 2,7 fois.[32]

2.6 conclusion :

Dans ce chapitre, nous avons donné un aperçu des méthodes de détection d'objets basées sur l'apprentissage en profondeur. Nous sommes partis du concept de découverte d'objets et de ses bénéfices et nous avons passé en revue les méthodes de détection d'objets les plus connues et les plus utilisées en examinant leurs structures SIFT, HOG, SSD, R-CNN...etc. Puis nous nous sommes appuyés sur la méthode YOLO. Finalement, un résumé de quelques travaux pertinents portant spécifiquement sur la détection des panneaux de signalisation.

CHAPITRE 3

APPROCHE PROPOSÉE

3.1 Introduction :

La détection d'objets consiste à identifier la présence et la position d'objets dans une image, ainsi que leur catégorie. Pour cela, on utilise des algorithmes qui dessinent une boîte englobante autour de l'objet détecté. Il existe différents types d'algorithmes, qui varient en termes de précision, de vitesse, de ressources requises et du nombre de classes supportées.

L'apprentissage en profondeur est une technique d'apprentissage automatique qui permet d'améliorer la précision de la détection d'objets. Cette technique repose sur des réseaux de neurones profonds, capables d'apprendre à détecter des motifs complexes dans les images. YOLO v5 est un exemple d'algorithme de détection d'objets basé sur l'apprentissage en profondeur. Il est connu pour sa rapidité et sa précision, ainsi que sa capacité à détecter un grand nombre de classes d'objets différents.

3.2 Deep Learning :

3.2.1 La définition de Deep Learning :

Les réseaux de neurones étaient jusqu'à récemment limités en complexité en raison de la puissance de calcul limitée disponible. Cependant, grâce aux progrès réalisés dans l'analyse du Big Data, les réseaux de neurones peuvent désormais être conçus plus vastes et sophistiqués. Les ordinateurs sont ainsi en mesure d'observer, d'apprendre et de réagir à des situations complexes plus rapidement que les humains. Le Deep Learning est une technologie qui a été très utile pour la classification d'images, la traduction de langues et la reconnaissance vocale. Elle est capable de résoudre tout problème de reconnaissance de formes sans intervention humaine.

En d'autres termes, le Deep Learning est une technologie puissante qui offre de nombreuses possibilités pour résoudre des problèmes complexes. Grâce aux avancées réalisées, les réseaux de neurones peuvent maintenant traiter de grandes quantités de données, ce qui leur permet d'offrir des performances plus élevées et une précision accrue. Les applications du Deep Learning sont vastes et peuvent être utilisées pour améliorer différents aspects de la vie quotidienne tels que la sécurité routière, la reconnaissance vocale, la traduction de langues et bien plus encore.[33]

3.2.2 L'importance de Deep Learning :

— Contrairement à l'apprentissage en profondeur, qui peut fonctionner avec des données structurées et non structurées, l'apprentissage automatique ne peut travailler qu'avec des

ensembles de données structurées ou semi-structurées.

— Les algorithmes de machine learning utilisent des ensembles de données étiquetés pour extraire des modèles, tandis que le deep learning traite de grandes quantités de données en entrée et utilise l'analyse de ces données pour extraire les caractéristiques d'un objet.

— La performance des algorithmes d'apprentissage automatique peut diminuer lorsqu'on augmente le nombre de données, ce qui nécessite l'utilisation de techniques d'apprentissage en profondeur pour maintenir la précision du modèle.[33]

3.2.3 Applications of deep learning :

Le Deep Learning est utilisé dans de nombreuses applications, telles que :[34]

• **1-La reconnaissance vocale :**conversion de la parole en texte, amélioration de la qualité audio, extraction de motifs sonores.

• **2-La vision par ordinateur :**reconnaissance d'images, de vidéos et de séquences animées, classification et interprétation, amélioration de la qualité.

• **3-Le traitement du langage naturel :** analyse de texte et de discours, reconnaissance de texte et de discours, traduction et génération de texte.

• **4-Les robots et l'automatisation :**amélioration des capacités des robots à interagir avec leur environnement et à apprendre de l'expérience, amélioration de la performance de l'automatisation dans les usines et les entreprises.

• **5-L'apprentissage automatique en médecine :**analyse d'images médicales et radiologiques, reconnaissance de maladies et diagnostics précis, analyse de données médicales.

• **6-Les systèmes intelligents :** développement de systèmes intelligents pour contrôler les appareils intelligents, amélioration de la performance de l'automatisation dans les maisons et les bureaux.

3.2.4 Types de modèles utilisant des architectures Deep Learning) :

Réseaux neuronaux convolutifs (CNN) :

Les CNN, également appelés ConvNets, sont composés de nombreuses couches qui sont responsables de l'analyse et de l'extraction de caractéristiques spécifiques des données. Ces réseaux de neurones convolutifs sont souvent utilisés pour la détection et l'analyse d'objets dans des images, telles que des images satellites ou des images médicales. Ils peuvent également être utilisés pour la détection d'anomalies ou la prédiction de séries chronologiques.[35]

Réseaux neuronaux récurrents (RNN) :

Les réseaux de neurones récurrents (RNN) possèdent des connexions qui forment des cycles dirigés, permettant aux sorties du RNN d'être utilisées comme entrées pour la phase suivante. Cette rétroaction permet à la mémoire interne du RNN de mémoriser les entrées précédentes. Les RNN sont couramment utilisés dans des applications telles que la génération de sous-titres pour les images, le traitement du langage naturel et la traduction automatique.[35]

Réseaux de mémoire à long et court terme (LSTM) :

Les LSTM, ou réseaux de neurones récurrents à mémoire à long terme, sont une forme spécifique de réseaux de neurones récurrents (RNN) qui peuvent apprendre et mémoriser les dépendances à long terme. La capacité à se souvenir des informations passées pendant de longues périodes est un comportement par défaut de ce type de réseau neuronal. Les LSTM sont particulièrement utiles dans la prédiction de séries chronologiques, car ils peuvent se souvenir des entrées précédentes et donc mieux comprendre les tendances temporelles. Les LSTM ont une structure en forme de chaîne où quatre couches interagissent de manière unique. En plus de la prédiction de séries chronologiques, les LSTM sont souvent utilisés pour la reconnaissance vocale, la composition musicale et le développement pharmaceutique.[35]

Réseaux de fonction de base radiale (RBFN) :

Les réseaux de fonctions de base radiales (RBFN) sont des réseaux de neurones artificiels qui utilisent des fonctions de base radiales comme fonctions d'activation. Ces réseaux sont composés d'une couche d'entrée, d'une couche cachée et d'une couche de sortie. Ils sont couramment utilisés pour la classification, la prédiction de séries temporelles et la régression linéaire. Les RBFN sont particulièrement utiles dans les cas où les données d'entrée sont complexes ou non linéaires, nécessitant une représentation spatiale en dimensions supérieures pour une meilleure classification.[35]

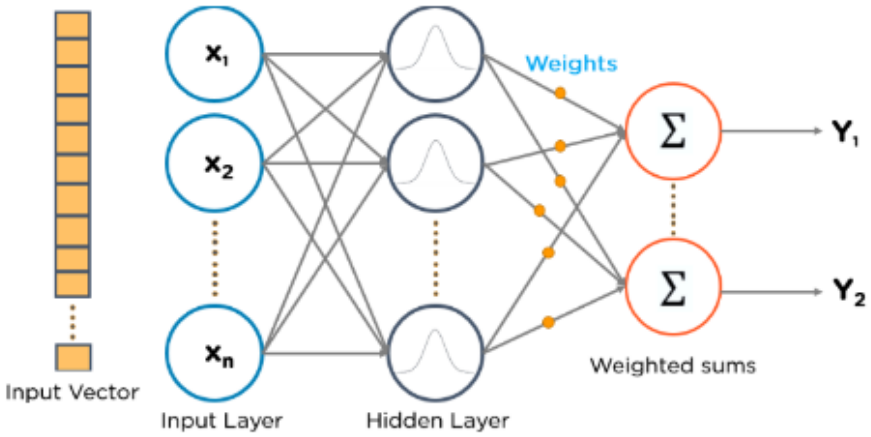

FIGURE 3.1 – exemple de RBFN.

Réseaux adversariaux génératifs (GAN) :

Les réseaux de neurones antagonistes génératifs (GAN) sont une technique de deep learning qui permet de créer de nouvelles données qui ressemblent à celles utilisées pour l'entraînement. Les GAN se composent d'un générateur et d'un discriminateur. Le générateur apprend à créer de nouvelles données en prenant en compte les données d'apprentissage. Si le générateur produit des données incorrectes, le discriminateur apprend à détecter ces erreurs. Les GAN sont souvent utilisés dans le domaine du jeu vidéo pour améliorer les textures 2D.[35]

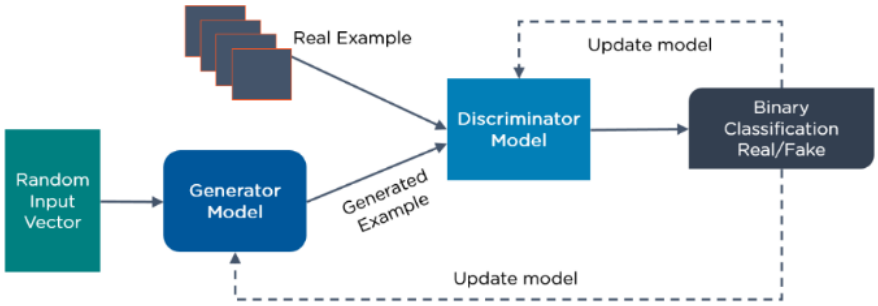

FIGURE 3.2 – exemple de GAN.

Machines de Boltzmann restreintes (RBM) :

Les réseaux neuronaux stochastiques à deux couches sont constitués d'unités visibles et cachées, et peuvent apprendre à partir d'une distribution de probabilité sur un ensemble d'entrées. Ces réseaux artificiels utilisent des techniques stochastiques pour ajuster les poids et les biais de leurs connexions, ce qui leur permet d'explorer différentes combinaisons de variables et d'améliorer leurs performances au fil du temps.[35]

3.3 Algorithme YOLO (You Only Look Once) :

3.3.1 Architecture :

L'algorithme YOLO (You Only Look Once) utilise une architecture de réseau de neurones convolutionnel (CNN) pour la détection d'objets dans une image. Cette architecture prend une image en entrée et la redimensionne en 448x448 tout en conservant le rapport d'aspect et en ajoutant un remplissage pour l'adapter à la taille requise. Le modèle CNN est composé de 24 couches de convolution, suivies de 4 couches de max-pooling et de 2 couches entièrement connectées. Pour simplifier la structure du réseau, une convolution de 1x1 est utilisée pour réduire le nombre de couches, suivie d'une convolution de 3x3. La fonction d'activation utilisée dans toute l'architecture est Leaky ReLU, sauf pour la couche de convolution où une fonction d'activation linéaire est appliquée.[30]

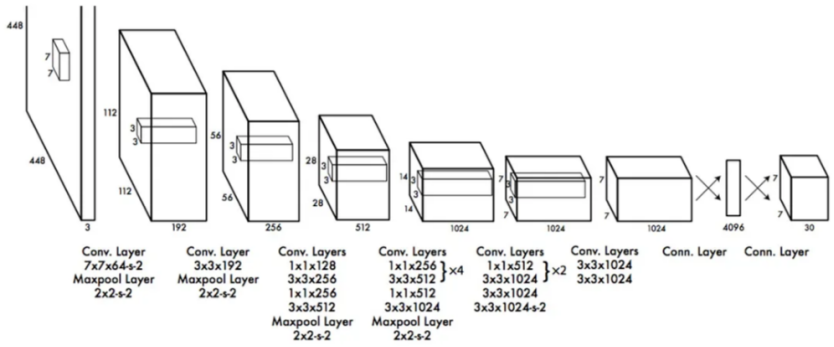

FIGURE 3.3 – Architecture l'algorithme yolo.

- **fonction d'algorithme :**

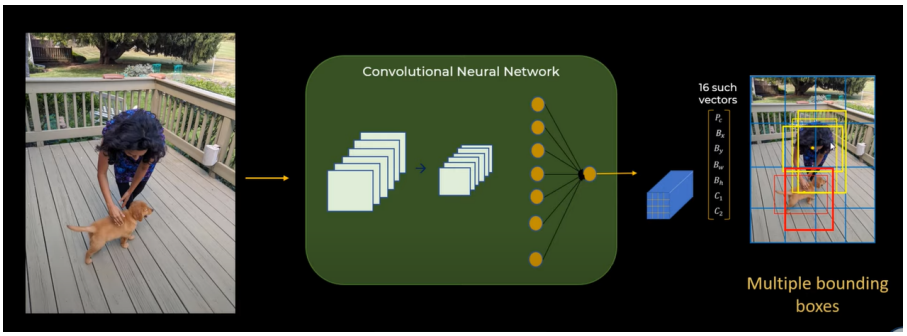

FIGURE 3.4 – fonction d'algorithme yolo.

Supposons que nous enquêtions sur un problème de classification d'images, où notre objectif est de déterminer si une image représente un chien ou une personne. Dans ce contexte, la sortie d'un réseau de neurones est binaire. Nous pouvons attribuer une valeur de 1 pour représenter

un chien et une valeur de 0 pour représenter une personne. De plus, nous pouvons également considérer des informations sur l'emplacement de l'objet dans l'image, telles que la boîte qui entoure l'objet, comme les éléments du vecteur de sortie sur le réseau de neurones.

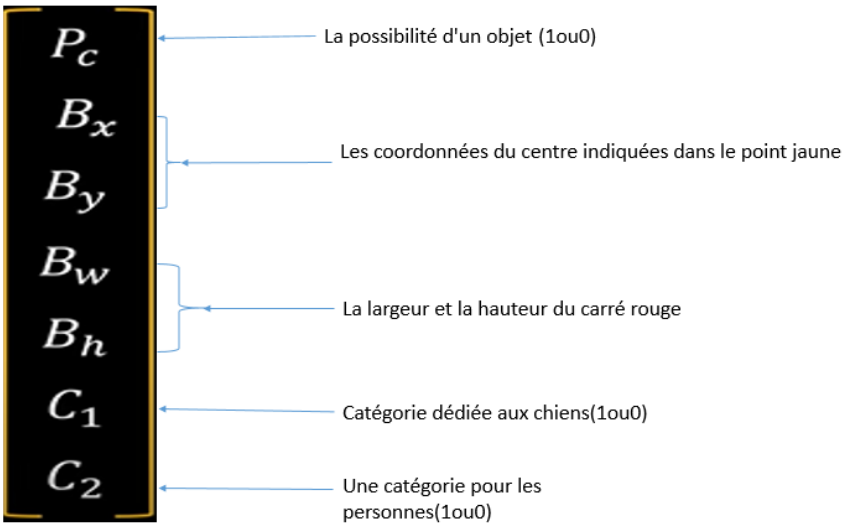

FIGURE 3.5 – Le vecteur de sortie sur le réseau de neurones.

Le modèle a connu 5 versions successives, chacune améliorant la précédente en termes de vitesse et de précision :

Le model yolo v2 :

Dans cette version, les auteurs Joseph Redmon et Ali Farhadi ont travaillé sur l'amélioration du modèle YOLO en proposant une version plus rapide et plus performante. Ils ont modifié la première architecture en utilisant Darknet-19 comme base et en augmentant le nombre de couches à 30 (contre 26 pour YOLO v1). De plus, des couches de normalisation par lots ont été ajoutées après chaque couche de convolution pour améliorer la précision. [36] La résolution de

Type	Filters	Size/Stride	Output
Convolutional	32	3 × 3	224 × 224
Maxpool		2 × 2/2	112 × 112
Convolutional	64	3 × 3	112 × 112
Maxpool		2 × 2/2	56 × 56
Convolutional	128	3 × 3	56 × 56
Convolutional	64	1 × 1	56 × 56
Convolutional	128	3 × 3	56 × 56
Maxpool		2 × 2/2	28 × 28
Convolutional	256	3 × 3	28 × 28
Convolutional	128	1 × 1	28 × 28
Convolutional	256	3 × 3	28 × 28
Maxpool		2 × 2/2	14 × 14
Convolutional	512	3 × 3	14 × 14
Convolutional	256	1 × 1	14 × 14
Convolutional	512	3 × 3	14 × 14
Convolutional	256	1 × 1	14 × 14
Convolutional	512	3 × 3	14 × 14
Maxpool		2 × 2/2	7 × 7
Convolutional	1024	3 × 3	7 × 7
Convolutional	512	1 × 1	7 × 7
Convolutional	1024	3 × 3	7 × 7
Convolutional	512	1 × 1	7 × 7
Convolutional	1024	3 × 3	7 × 7
Convolutional	1000	1 × 1	7 × 7
Avgpool		Global	1000
Softmax			

FIGURE 3.6 – Architecture de darknet19.

l'image pour la détection a également été augmentée à 448, avec une grille de stride=32 et la prédiction de 5 boîtes limitantes pour chaque cellule. Les boîtes d'ancrage ont été introduites.

Le model yolo v3 :

En 2018, Joseph Redmon et Ali Farhadi ont amélioré progressivement la version précédente de YOLO en utilisant Darknet53 comme backbone pour l'extraction de caractéristiques. Ils ont également utilisé une régression logistique pour calculer un score d'objectivité pour chaque boîte de délimitation et la perte d'entropie croisée binaire pour les prédictions de classe. La version YOLOv2 avait des problèmes pour détecter les petits objets, mais dans cette version, la représentation des boîtes a été faite à trois échelles différentes pour améliorer la détection de ces objets.[37]

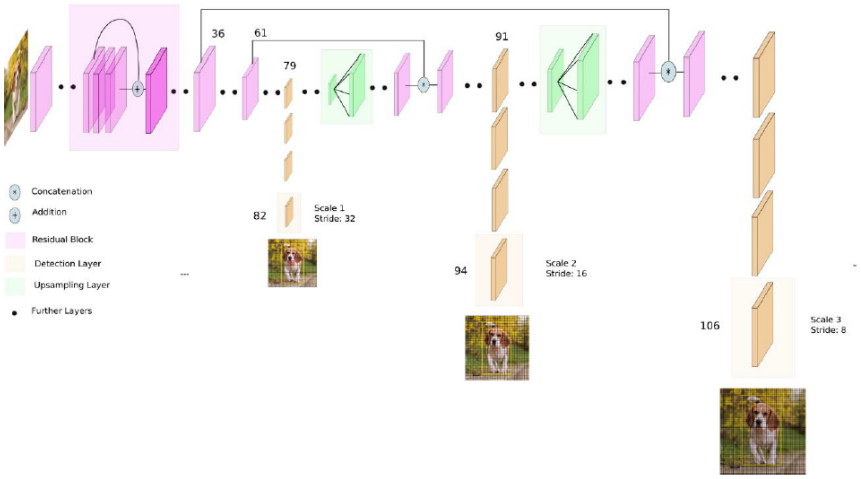

FIGURE 3.7 – Architecture de model yolo v3.

Le model yolo v4 :

L'architecture de YOLOv4 se compose de trois parties principales : le backbone, le neck et la tête. Le backbone est basé sur CSPDarknet53, qui est une version améliorée de Darknet53, utilisé dans YOLOv3. Le neck est composé de deux modules : SPP et PANet. SPP (Spatial Pyramid Pooling) est utilisé pour capturer les informations de contexte à différentes échelles, tandis que PANet (Path Aggregation Network) fusionne les fonctionnalités à différentes résolutions pour une meilleure précision de la détection. Enfin, la tête est identique à celle de YOLOv3 et utilise des boîtes d'ancrage pour prédire les coordonnées des objets ainsi que les scores de confiance et les classes. La figure ci-jointe illustre l'architecture de YOLOv4 en d étail.[38]

YOLO v4 utilise deux approches différentes pour améliorer la précision de la détection des objets. La première, appelée Bag of Freebies (BoF), consiste en des techniques qui modifient la stratégie de formation du modèle pour améliorer sa performance. La seconde, appelée Bag of Specials (BoS), consiste en des modules et des méthodes de post-traitement qui peuvent

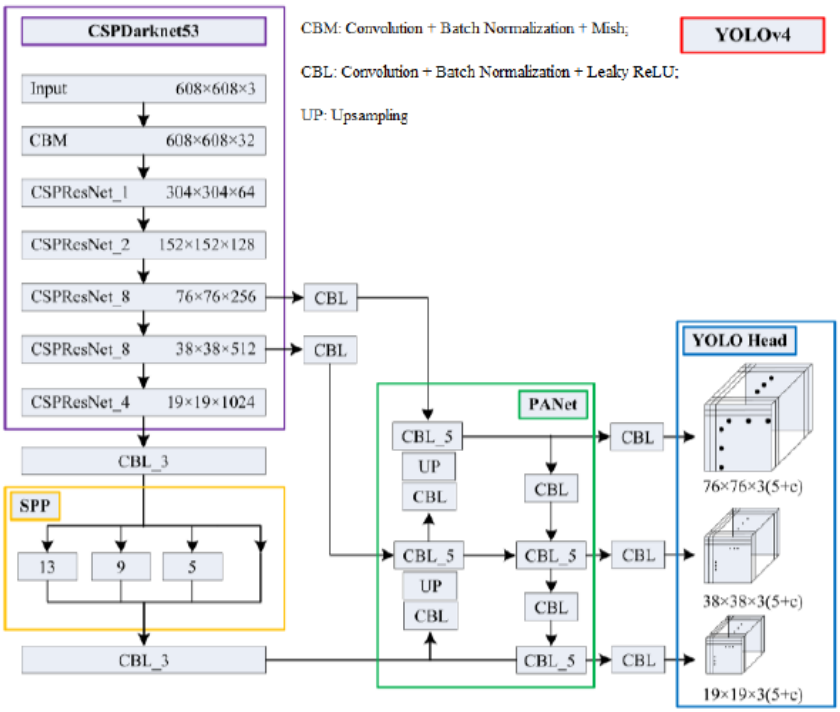

FIGURE 3.8 – Architecture de model yolo v4.

augmenter le coût de l'inférence, mais qui améliorent également la précision de la détection des objets.

Le model yolo v5 :

Le modèle en question suscite des controverses car, à ce jour, l'auteur Glenn Jocher n'a pas publié de documents permettant à la communauté scientifique d'évaluer son benchmark. De plus, il ne semble pas avoir introduit de nouvelles techniques qui justifieraient sa prétention à être la prochaine version de YOLO. En réalité, ce modèle est davantage considéré comme une extension de PyTorch pour YOLOv3.[38]

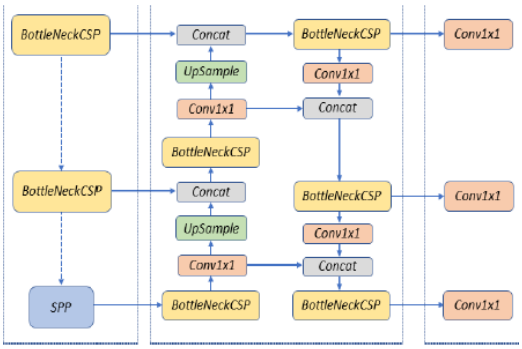

FIGURE 3.9 – Architecture de model yolo v5.

- **Le backbone de YOLOv5 :** se sert de CSPDarknet pour extraire les caractéristiques des

Model yolo	Dataset	Map	Size	FPS
Yolov1	Pascalvoc 07+12	63.4%	448	45
Yolov2	Pascalvoc 07+12	78.6%	544	40
Yolov3	MS COCO	33.0%	608	20
Yolov4	MS COCO	43.5%	608	23
Yolov5-x	MS COCO	50.4%	640	62.5

TABLE 3.1 – les résultats de différentes versions YOLO.

images. CSPDarknet est un réseau partiel multicouche qui permet de réaliser cette extraction à différents niveaux.

• **Dans YOLOv5, le Neck :** utilise PANet pour construire un réseau de pyramides de caractéristiques qui agrège les informations et les transmet à la tête de réseau pour effectuer la prédiction.[38]

• **Le Head de YOLOv5 :** est constitué de couches qui produisent des prédictions basées sur les boîtes d'ancrage pour la détection d'objets.

• **Activation et optimisation :** YOLOv5 applique des fonctions d'activation telles que la sigmoïde et la ReLU fuyante, et offre des options d'optimisation comme SGD et ADAM.

• **La fonction de perte :** utilisée par YOLOv5 est l'entropie croisée binaire, avec une perte logits.

-le Tableau présente des résultats comparatifs entre plusieurs versions de YOLO.[38]

3.3.2 Avantages et inconvinients de YOLO :

1-Avantages : L'algorithme YOLO (You Only Look Once) est un algorithme de reconnaissance d'objets en temps réel qui a été largement utilisé pour la détection d'objets dans des images et des vidéos. Voici quelques-uns des avantages de l'algorithme YOLO[39] :

• **Vitesse :** L'un des principaux avantages de YOLO est sa rapidité. Il utilise une approche de détection d'objets en une seule étape, ce qui signifie qu'il n'a besoin que d'un seul passage pour détecter les objets dans une image ou une vidéo. Par conséquent, il est considérablement plus rapide que les approches de détection d'objets en deux étapes.

• **Précision :**Bien que YOLO soit très rapide, il est également très précis. Il utilise une architecture de réseau de neurones profonds qui lui permet de détecter avec précision les objets dans une image ou une vidéo.

• **Détection multi-objets :** YOLO peut détecter plusieurs objets dans une seule image ou une seule vidéo en une seule fois. Il est capable de détecter plusieurs objets de différentes tailles et formes, et il peut même détecter des objets qui se chevauchent.

• **Facilité d'utilisation :** YOLO est relativement facile à utiliser. Il a une mise en œuvre simple et peut être facilement adapté à différentes tâches de détection d'objets.

• **Disponibilité des ressources :** Il existe de nombreuses ressources disponibles en ligne pour aider les utilisateurs à utiliser YOLO, y compris des tutoriels et des exemples de code.

2-inconvénients : Bien que l'algorithme YOLO (You Only Look Once) ait de nombreux avantages, il présente également quelques inconvénients. Voici quelques-uns des inconvénients de YOLO [39] :

• **Précision limitée pour les petits objets :** Bien que YOLO soit très précis pour les objets de taille moyenne à grande, sa précision peut être limitée pour les petits objets. En effet, les petits objets peuvent être difficiles à détecter car ils ont une faible résolution et sont souvent cachés par d'autres objets.

• **Sensibilité aux changements d'échelle :** YOLO peut être sensible aux changements d'échelle. Cela signifie que si les objets que vous essayez de détecter ont des tailles différentes, vous devrez peut-être entraîner plusieurs modèles pour chaque échelle.

• **Besoin de données d'entraînement massives :** Pour obtenir les meilleurs résultats de détection, il est important d'avoir de grandes quantités de données d'entraînement. Cela peut prendre beaucoup de temps et nécessiter des ressources informatiques importantes pour entraîner le modèle.

• **ensible aux perturbations :**YOLO peut être sensible aux perturbations dans les images, telles que les ombres, les réflexions ou les objets partiellement cachés. Cela peut conduire à des erreurs de détection ou à des faux positifs.

• **Faible résolution de détection :** YOLO peut avoir une résolution de détection inférieure à celle des approches de détection d'objets en deux étapes, ce qui peut entraîner une précision de détection inférieure.

3.4 Génération de l'ensemble de données sdv dataset :

En utilisant la technique de détection d'objets saillants basée sur la divergence des couleurs, représentée à l'aide d'un cube RVB. Le cube RVB représente les couleurs de l'image en trois dimensions : rouge, vert et bleu. La technique utilise des écarts-types pour identifier et afficher la zone d'uniformité de couleur possible correspondant à l'objet. Pour ce faire, nous calculons

la matrice de distance euclidienne entre chaque pixel de couleur RVB de l'image d'origine et les huit sommets du cube RVB. Pour chaque pixel, nous calculons huit distances (entre le pixel et le coin du cube), puis calculons l'écart type de ces distances, ce qui donne une valeur unique notée sdv_p. Le but de la représentation du cube RVB est d'afficher les classes de couleurs proches de l'homogénéité, c'est vraisemblablement l'objet. La technique proposée se concentre sur l'idée de représenter le cube de couleur RVB de l'image fournie en entrée. À cette fin, les

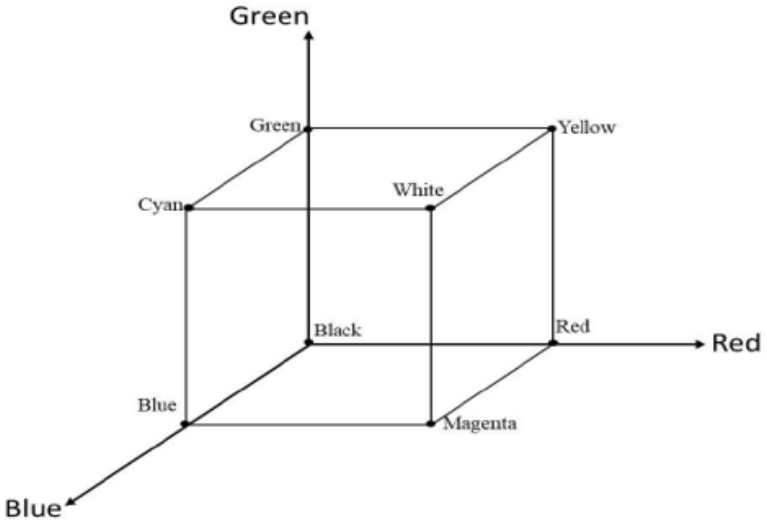

FIGURE 3.10 – Un cube 3D représentant géométriquement le RVB espace colorimétrique.

distances entre chaque pixel et les huit sommets du cube sont calculées à l'aide de l'équation 1. Le vecteur de distance d'un pixel donné p est défini comme $D_p = [d0, d1, d2, d3, d4, d5, d6, d7]$, où di correspond à la distance entre le pixel p et le sommet i (i= 0 . . . 7) du cube qui se calcule comme suit[40] :

$$d_i = \sqrt{r_i^2 + g_i^2 + b_i^2}$$

Où r_i, g_i et b_i représentent respectivement le rouge, le vert et le bleu. La valeur d'un pixel p dépend d'un sommet i. L'écart type de ces distances, donne une valeur unique à chaque pixel notée sdv_p ce qui conduit à réduire la taille de l'image tout en préservant ses propriétés pertinentes.

$$sdv_p = \frac{\sqrt{\Sigma(d_i - averageD_p)^2}}{8}$$

où di est égal à la distance entre le pixel p Les sommets du cube i(i = 0 . . . 7). $averageD_p$ est la moyenne du vecteur de distance D_p.

Nous définissons le coefficient f pour séparer les pixels symétriques. Le cube qui reflète la couleur est représenté par un nouveau cube aux bords irréguliers. La valeur de di est calculée comme suit[40] :

$$d_i = f_i * \sqrt{r_i^2 + g_i^2 + b_i^2}$$

Nous attribuons donc à chacun des sommets de cube des valeurs f différentes. Le rouge se voit attribuer une valeur de 150 pour indiquer un danger tandis que une valeur de 100 au jaune pour indiquer la prudence. Alors que le bleu se voit attribuer une valeur de 50 pour indiquer les signaux obligatoires. Quant au blanc une valeur de 50, car la plupart des signaux ont un fond blanc.

3.5 Architecture de l'approche proposée :

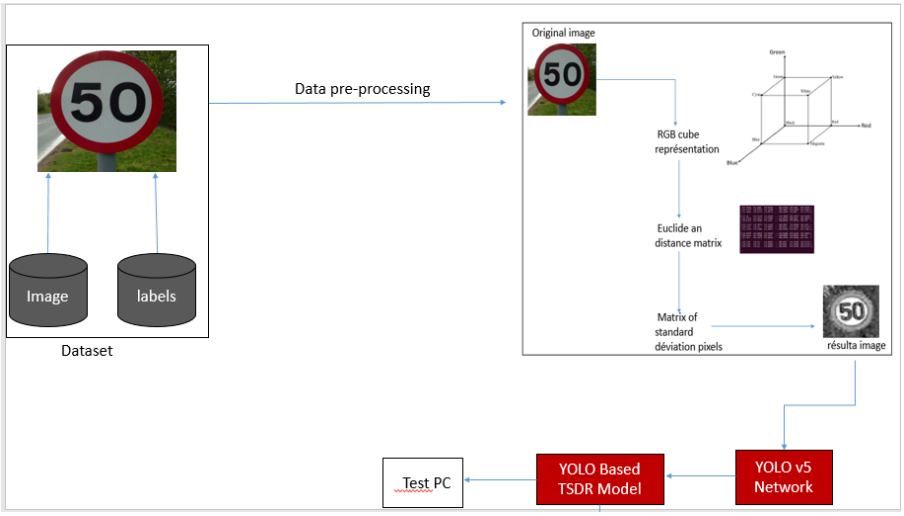

FIGURE 3.11 – Architecture générale.

Ce travail suit le schéma de détection en temps réel et reconnait les panneaux de signalisation et les objets routiers sur les images de scène naturelles. Nous commençons par construire de l'ensemble de données 'sdv dataset' à partir de l'ensemble des données d'origine en suivant les étapes mentionnées dans la section précédente. A savoir, pour chaque image :

- RVB cube représentation de l'image originale.

- Calcule de la matrice de distance euclidienne.
- Calcule de la matrice de l'équart type à partir de de la matrice de distance euclidienne.

3.6 Conclusion :

Dans ce chapitre, nous avons fourni une explication détaillée de l'apprentissage en profondeur et de l'algorithme YOLO, ainsi que de leur application dans la détection de panneaux routiers. Nous avons commencé par une définition générale de l'apprentissage en profondeur, en expliquant son importance, ses applications et les différents types de modèles existants. Ensuite, nous avons présenté l'algorithme YOLO et expliqué sa structure et les différences entre chaque version. Nous avons mis l'accent sur la version YOLO V5 utilisée dans notre conception et présenté une explication générale de la phase de prétraitement.

CHAPITRE 4

IMPLÉMENTATION ET RÉSULTATS

4.1 Introduction :

Dans ce chapitre, nous exposons les moyens employés pour créer et mettre en place notre système, ainsi que les normes appliquées pour extraire les caractéristiques avec YOLO et la méthode utilisée pour la classification. Nous évaluons la validité de notre travail en détaillant les différentes expériences menées et les résultats obtenus.

4.2 Environnement de développement :

4.2.1 Environnement matériels(Hardware) :

Dans cette étude, nous présentons des expériences détaillées ainsi que des évaluations pour mesurer l'efficacité de notre modèle. Nous avons utilisé un ensemble de données d'images contenant des panneaux de signalisation, appelé "GTSRB Dataset"2011, comme base pour nos expériences. Toutes les expériences ont été effectuées sur un ordinateur DELL équipé d'un processeur Intel(R) Core(TM) i5-6200U CPU @ 2.30GHz (4 CPUs), de 8 Go de RAM .

4.2.2 Environnement logiciel(Software) :

• **Python :** Python est un langage de programmation de haut niveau et interprété qui convient à une large gamme d'applications. Il a été créé par Guido van Rossum en 1991 et est particulièrement connu pour sa lisibilité grâce à l'utilisation d'espaces significatifs dans sa philosophie de conception. Python offre des constructions pour une programmation claire et évolutive, ainsi qu'un système de typage dynamique et une gestion automatique de la mémoire.

Il prend en charge plusieurs paradigmes de programmation tels que l'orienté objet, l'impératif, le fonctionnel et le procédural, ainsi qu'une bibliothèque standard complète et étendue. Python est un langage open-source qui fonctionne sur de nombreux systèmes d'exploitation, ce qui en fait un langage de programmation diversifié et puissant.[41]

FIGURE 4.1 – Logo Python.

Python est également facile à apprendre avec une syntaxe simple et encourageante pour

les débutants, tout en offrant des fonctionnalités avancées pour les utilisateurs expérimentés. Il est également doté de bibliothèques populaires pour l'intelligence artificielle telles que Keras, Tensorflow, Numpy, Pandas, OpenCV, Pytorch, etc.

• **OpenCV :** OpenCV (Open Source Computer Vision Library) est une bibliothèque logicielle open source destinée à la vision par ordinateur et à l'apprentissage automatique. Elle a été créée dans le but de fournir une infrastructure commune pour les applications de perception artificielle, et d'accélérer l'adoption de cette technologie dans les produits commerciaux. Sous licence BSD, OpenCV permet aux entreprises de facilement utiliser et modifier son code. La bibliothèque contient plus de 2500 algorithmes optimisés, couvrant un large éventail d'applications, de la reconnaissance faciale à la détection d'objets, en passant par la classification d'actions humaines dans les vidéos, la stabilisation de caméra, le suivi d'objets en mouvement, l'extraction de modèles 3D, etc.[42]

FIGURE 4.2 – logo open cv.

• **TensorFlow :** TensorFlow est une plate-forme open source complète pour l'apprentissage automatique qui offre un écosystème flexible de bibliothèques, d'outils et de ressources communautaires. Il permet aux chercheurs de faire progresser l'état de l'art en ML, tout en permettant aux développeurs de créer et de déployer facilement des applications alimentées par ML.

FIGURE 4.3 – Logo Tensorflow.

Dans l'ensemble, TensorFlow fournit une solution de bout en bout pour toutes les étapes du processus d'apprentissage automatique, de la préparation des données à la formation et au déploiement des modèles.[43]

• **Keras :** Keras est une API de réseau neuronal de haut niveau open source, écrite en Python et capable de s'exécuter sur divers frameworks de niveau inférieur tels que TensorFlow,

Theano et Microsoft Cognitive Toolkit (CNTK). Keras fournit une interface conviviale pour la création et la formation de modèles d'apprentissage en profondeur et permet aux utilisateurs de prototyper et d'expérimenter rapidement différentes architectures de réseaux neuronaux. Il est

FIGURE 4.4 – Logo Keras.

conçu pour être modulaire, flexible et extensible, ce qui facilite l'ajout de nouvelles couches, de fonctions de perte et d'autres composants au réseau neuronal. Keras prend également en charge les réseaux de neurones convolutionnels (CNN) et les réseaux de neurones récurrents (RNN) et dispose d'un support intégré pour les tâches d'apprentissage en profondeur courantes telles que la classification d'images, la détection d'objets et le traitement du langage naturel (NLP). Dans l'ensemble, Keras est un choix populaire pour les chercheurs et les développeurs travaillant dans le domaine de l'apprentissage en profondeur, en raison de sa facilité d'utilisation, de sa flexibilité et de sa capacité à tirer parti de puissantes ressources informatiques telles que les GPU.[44]

• **Matplotlib :** Matplotlib est une bibliothèque open source de visualisation de données en Python. Elle permet de créer une grande variété de graphiques et de visualisations, allant des simples diagrammes en barres aux graphiques complexes en 3D. Matplotlib est largement utilisé dans les domaines de la science des données, de la recherche, de la visualisation de données et de l'apprentissage automatique.

La bibliothèque Matplotlib offre une grande flexibilité pour personnaliser les graphiques. Elle permet de contrôler minutieusement les éléments tels que les axes, les étiquettes, les couleurs, les styles de ligne et les légendes. Matplotlib offre également la possibilité de sauvegarder les graphiques dans différents formats d'image, tels que PNG, JPEG, PDF, SVG, etc.

Matplotlib s'intègre bien avec les autres bibliothèques de calcul scientifique en Python, telles que NumPy et Pandas, ce qui en fait un outil puissant pour visualiser et explorer les données. Il est également possible d'interagir avec les graphiques Matplotlib de manière interactive, par exemple en zoomant, en déplaçant ou en sélectionnant des points sur le graphique.[45]

• **PyTorch :** PyTorch est une bibliothèque open source de calcul tensoriel et d'apprentissage automatique basée sur le langage de programmation Python. Elle est principalement utilisée pour la création de réseaux de neurones profonds et pour l'implémentation d'algorithmes

d'apprentissage automatique.

PyTorch est connu pour sa flexibilité et sa facilité d'utilisation, ce qui en fait un choix populaire parmi les chercheurs et les praticiens en apprentissage automatique. Il offre une grande variété de fonctionnalités, y compris des outils pour la création et l'entraînement de réseaux de neurones, la manipulation des données, le calcul parallèle sur GPU, la gestion automatique des gradients pour le rétropropagation et la mise en œuvre d'architectures de réseaux complexes.

Une caractéristique clé de PyTorch est son système de suivi des gradients dynamiques. Cela signifie que les opérations effectuées sur les tenseurs peuvent être suivies et les gradients peuvent être calculés automatiquement, ce qui facilite la mise en œuvre de modèles d'apprentissage automatique complexes et la réalisation de tâches telles que l'optimisation et la rétropropagation.[46]

FIGURE 4.5 – Logo pyTorch.

• **Pandas :** Pandas est une bibliothèque open source de manipulation et d'analyse de données pour Python. Il offre un moyen flexible et performant de travailler avec des données structurées, y compris des données tabulaires telles que des feuilles de calcul et des tables SQL. Pandas est construit sur NumPy et fournit un certain nombre de structures de données utiles, y compris DataFrame et Series, qui permettent une manipulation, un nettoyage et une transformation efficaces des données. Pandas comprend également un large éventail de fonctions pour le filtrage, l'agrégation et l'analyse statistique des données, ce qui en fait un outil puissant pour l'analyse et la modélisation exploratoire des données. De plus, Pandas s'intègre bien avec d'autres bibliothèques Python telles que Matplotlib et Scikit-learn, ce qui en fait un outil clé dans le workflow de science des données et d'apprentissage automatique.[47]

• **NumPy :** NumPy (Numeric Python) est une bibliothèque open source très populaire en Python, qui offre un support puissant pour la manipulation de tableaux multidimensionnels et des opérations mathématiques sur ces tableaux. Elle constitue une base essentielle pour de nombreuses autres bibliothèques scientifiques et d'apprentissage automatique en Python.

NumPy permet de créer, manipuler et effectuer des opérations efficaces sur des tableaux de données de n'importe quelle dimension. Les tableaux NumPy, appelés "ndarrays" (n-dimensional arrays), offrent une structure de données homogène et optimisée pour les calculs numériques.

La bibliothèque NumPy fournit une large gamme de fonctions mathématiques et statistiques pour effectuer des opérations telles que l'addition, la soustraction, la multiplication, la division,

le calcul de la moyenne, l'écart-type, etc. sur des tableaux de données. Elle permet également d'effectuer des opérations vectorisées, ce qui signifie que les opérations sont appliquées à l'ensemble du tableau plutôt qu'aux éléments individuels, ce qui améliore les performances et la lisibilité du code.[48]

NumPy offre également des fonctionnalités avancées pour la manipulation et la transformation des tableaux, telles que le découpage (slicing), la mise en forme (reshaping), la permutation, l'indexation avancée, etc. Ces fonctionnalités permettent de sélectionner et de manipuler facilement les éléments des tableaux, ainsi que de réaliser des opérations de traitement des données efficaces.

FIGURE 4.6 – Logo NumPy .

• **Colab :**"Colab" est un terme abrégé pour "Colaboratory", qui est une plate-forme Web développée par Google Research qui permet aux utilisateurs d'écrire et d'exécuter du code Python dans un navigateur Web, avec un accès à des ressources informatiques gratuites basées sur le cloud telles que des GPU et TPU.

FIGURE 4.7 – Logo colab.

Colab est conçu pour être convivial et accessible à tous, sans qu'aucune configuration ou installation ne soit nécessaire. Il est particulièrement utile pour l'apprentissage automatique, l'analyse de données et l'éducation, car il offre un moyen simple d'écrire et d'exécuter du code en collaboration et de partager les résultats avec d'autres. Colab est construit sur l'environnement de bloc-notes Jupyter et fournit un certain nombre de fonctionnalités et d'intégrations supplémentaires avec d'autres services Google.[49]

• **roboflow :**Roboflow est une plateforme en ligne conçue pour faciliter le processus de préparation et d'annotation des données pour l'apprentissage automatique et la vision par

ordinateur. Elle offre un ensemble d'outils et de fonctionnalités pour aider les chercheurs, les développeurs et les équipes de données à gérer, annoter, augmenter et préparer leurs ensembles de données.

La plateforme Roboflow prend en charge différents formats de données, tels que les images, les vidéos et les annotations, et offre des fonctionnalités d'annotation automatisée ainsi que des outils pour l'annotation manuelle. Elle permet aux utilisateurs de créer rapidement des jeux de données étiquetés pour l'entraînement de modèles d'apprentissage automatique.

Roboflow propose également des fonctionnalités d'augmentation des données, qui permettent de générer de nouvelles variations des images existantes en appliquant des transformations telles que le recadrage, la rotation, le redimensionnement, etc. Cela permet d'enrichir les ensembles de données et d'améliorer les performances des modèles d'apprentissage automatique.

En outre, Roboflow intègre des outils de gestion des versions, qui permettent de suivre et de gérer les modifications apportées aux ensembles de données au fil du temps. Cela facilite la collaboration au sein des équipes et garantit la reproductibilité des expériences.[50]

FIGURE 4.8 – Logo roboflow.

• **Visual Studio Code :**Visual Studio Code est un éditeur de code source gratuit, open source et multiplateforme développé par Microsoft. Il fournit un environnement convivial et léger pour écrire et déboguer du code dans un large éventail de langages de programmation, notamment Python, JavaScript, C++ et bien d'autres. Visual Studio Code offre une variété de fonctionnalités, notamment la complétion de code intelligente, la coloration syntaxique, la mise en forme du code et les capacités de débogage. Il prend également en charge les extensions, qui permettent aux utilisateurs de personnaliser et d'étendre les fonctionnalités de l'éditeur en fonction de leurs besoins spécifiques. Visual Studio Code peut être exécuté sur les systèmes d'exploitation Windows, macOS et Linux, et peut être utilisé pour un large éventail de tâches de développement, notamment le développement Web, le développement d'applications mobiles et le développement de jeux.[51]

FIGURE 4.9 – Logo Visual Studio Code.

4.3 implémentation :

4.3.1 data set :

L'ensemble de données German Traffic Sign Recognition Benchmark (GTSRB) est un ensemble de données largement utilisé pour développer et évaluer des algorithmes de reconnaissance des panneaux de signalisation. Il est utilisé pour former et tester des modèles de vision par ordinateur pour reconnaître et classer les panneaux de signalisation que l'on trouve couramment sur les routes allemandes.

L'ensemble de données GTSRB se compose de plus de 50 000 images de panneaux de signalisation, capturées à partir de scènes du monde réel. L'ensemble de données couvre 43 classes différentes de panneaux de signalisation, compris les limites de vitesse, les panneaux directionnels, les panneaux d'avertissement, etc. Chaque image du jeu de données est étiquetée avec la classe correspondante du panneau de signalisation qu'elle représente.

Les images du jeu de données GTSRB varient en résolution, conditions d'éclairage, conditions météorologiques et autres facteurs pour simuler des scénarios réels. Cette diversité rend l'ensemble de données difficile et adapté à l'évaluation des performances des algorithmes de reconnaissance des panneaux de signalisation dans différentes conditions. Nous avons apporté quelques modifications au data set à cause de la RAM.Les résultats sont les suivants :[52]

	Train	Valide	Total	Classes	Acceleration
images	8600	1000	9600	29	93%
labels	8600	1000	9600		

TABLE 4.1 – Modifications du data set.

FIGURE 4.10 – L'échantillon aléatoire ci-dessus est typique de data set.

4.3.2 Explication du code :

Csv : La première étape consiste à convertir le fichier CSV en un groupe de fichiers TXT (convertir labels.csv en labels.txt).

```
import csv
with open('Train.csv', 'r') as csvfile:
    csvreader = csv.reader(csvfile)
    next(csvreader)  # skip the first row
    for row in csvreader:


        print(row)  # Debugging statement

        # Extract the relevant data from the row
        width = int(row[0])
        height = int(row[1])
        x1 = int(row[2])
        y1 = int(row[3])
        x2 = int(row[4])
        y2 = int(row[5])
        class_id = int(row[6])
        path = row[7]

        x_center = (x1 + x2) / 2 / width
        y_center = (y1 + y2) / 2 / height
        obj_width = (x2 - x1) / width
        obj_height = (y2 - y1) / height
        # Construct the filename
        filename = path.split('/')[-1].split('.')[0] + '.txt'

        # Write the content to a new text file
        with open(filename, 'w') as outfile:
            outfile.write('%s %s %s %s %s\n' % (class_id, x_center, y_center, obj_width, obj_height))
```

FIGURE 4.11 – convert labels.csv to labels.txt.

Le résultat de labels.txt :

FIGURE 4.12 – labels.txt.

Téléchargez data set sur roboflow pour vérifier que chaque image a ses labels étiquettes :

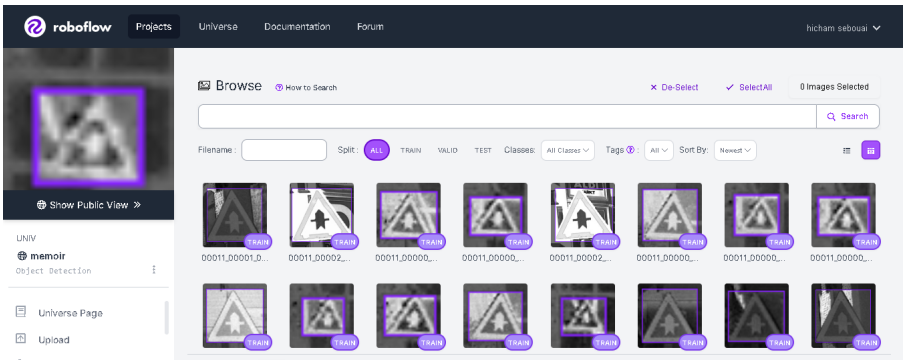

FIGURE 4.13 – vérifier les images.

Pour entraîner data set, nous utilisons le site Web de Colab :

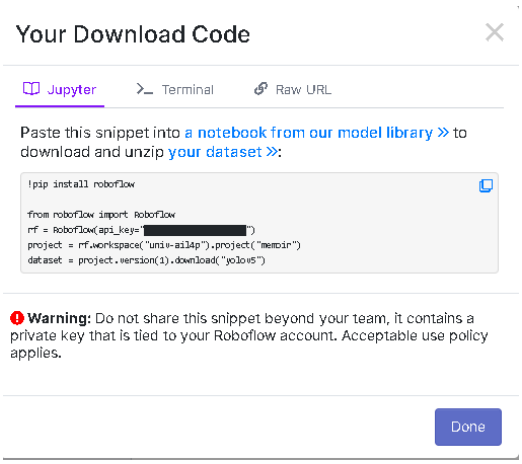

FIGURE 4.14 – formation par colab.

Le lien entre data set et le modèle se fait en plaçant le lien de l'image d'entraînement dans le train et le lien de l'image de validation dans Val et en plaçant le fichier dans le dossier de

données du projet :

FIGURE 4.15 – lier datasat et modal.

processus de train :

FIGURE 4.16 – train Data set.

4.3.3 Model proposé :

Génération de SDV dataset (standard deviation step) :

le code de sdv(preprocessing) :

- Le coefficient 150 pour le sommet rouge du cube se pour indiquer un danger.
- tandis que un coefficient de 100 au sommet jaune pour indiquer la prudence.
- Alors que le bleu se voit attribuer un coefficient de 50 pour indiquer les signaux obligatoires.
- Quant au blanc une valeur de 50, car la plupart des signaux ont un fond blanc.

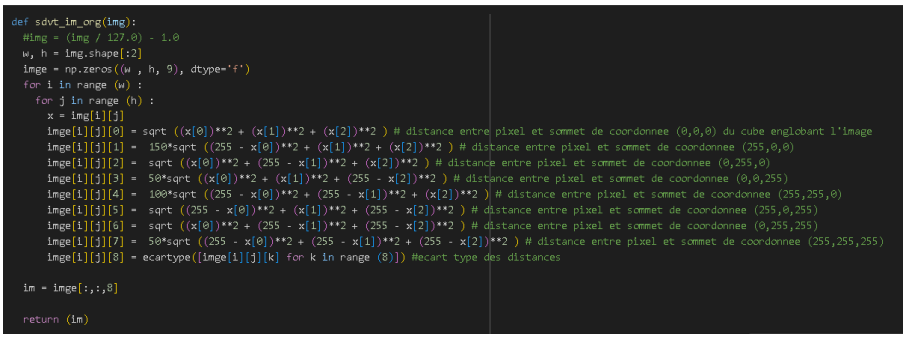

FIGURE 4.17 – Couche SDV.

YOLO(You Only Look Once) :

Dans le projet YOLOv5, la classe TransformerLayer est utilisée comme partie du réseau de base (backbone). Elle est responsable d'introduire des mécanismes d'auto-attention et des transformations non linéaires pour traiter les cartes de caractéristiques d'entrée.

```python
class TransformerLayer(nn.Module):
    # Transformer Layer https://arxiv.org/abs/2010.11929 (LayerNorm layers removed for better performance)
    def __init__(self, c, num_heads):
        super().__init__()
        self.q = nn.Linear(c, c, bias=False)
        self.k = nn.Linear(c, c, bias=False)
        self.v = nn.Linear(c, c, bias=False)
        self.ma = nn.MultiheadAttention(embed_dim=c, num_heads=num_heads)
        self.fc1 = nn.Linear(c, c, bias=False)
        self.fc2 = nn.Linear(c, c, bias=False)

    def forward(self, x):
        x = self.ma(self.q(x), self.k(x), self.v(x))[0] + x
        x = self.fc2(self.fc1(x)) + x
        return x
```

FIGURE 4.18 – Couche transtormerLayers.

Dans le projet YOLOv5, la classe BottleneckCSP est un composant clé utilisé dans le réseau de base (backbone). Elle est responsable de la création de blocs de bottleneck avec des connexions partielles trans-étapes (CSP) afin de faciliter le flux d'informations et d'améliorer les performances du réseau.

```python
class BottleneckCSP(nn.Module):
    # CSP Bottleneck https://github.com/WongKinYiu/CrossStagePartialNetworks
    def __init__(self, c1, c2, n=1, shortcut=True, g=1, e=0.5):  # ch_in, ch_out, number, shortcut, groups, expansion
        super().__init__()
        c_ = int(c2 * e)  # hidden channels
        self.cv1 = Conv(c1, c_, 1, 1)
        self.cv2 = nn.Conv2d(c1, c_, 1, 1, bias=False)
        self.cv3 = nn.Conv2d(c_, c_, 1, 1, bias=False)
        self.cv4 = Conv(2 * c_, c2, 1, 1)
        self.bn = nn.BatchNorm2d(2 * c_)  # applied to cat(cv2, cv3)
        self.act = nn.SiLU()
        self.m = nn.Sequential(*(Bottleneck(c_, c_, shortcut, g, e=1.0) for _ in range(n)))

    def forward(self, x):
        y1 = self.cv3(self.m(self.cv1(x)))
        y2 = self.cv2(x)
        return self.cv4(self.act(self.bn(torch.cat((y1, y2), 1))))
```

FIGURE 4.19 – Couche BottleneckCSP.

Dans YOLOv5, la classe Conv représente une couche de convolution utilisée dans l'architecture du modèle. Elle est responsable d'appliquer une opération de convolution au tenseur d'entrée.

```python
class Conv(nn.Module):
    # Standard convolution with args(ch_in, ch_out, kernel, stride, padding, groups, dilation, activation)
    default_act = nn.SiLU()  # default activation

    def __init__(self, c1, c2, k=1, s=1, p=None, g=1, d=1, act=True):
        super().__init__()
        self.conv = nn.Conv2d(c1, c2, k, s, autopad(k, p, d), groups=g, dilation=d, bias=False)
        self.bn = nn.BatchNorm2d(c2)
        self.act = self.default_act if act is True else act if isinstance(act, nn.Module) else nn.Identity()

    def forward(self, x):
        return self.act(self.bn(self.conv(x)))

    def forward_fuse(self, x):
        return self.act(self.conv(x))
```

FIGURE 4.20 – Couche convolution.

4.4 Evaluation :

Le résultat de train : Résultats des tests du programme : Les chercheurs ont fait ce projet

Class	Images	Instances	P	R	mAP50	mAP50-95: 100% 59/59 [00:24<00:00, 2.37it/s]
all	1884	1886	0.965	0.937	0.927	0.785
-Road narrows on right	1884	54	1	0.996	0.995	0.888
50 mph speed limit	1884	62	0.432	0.981	0.488	0.402
Attention Please-	1884	117	1	0.999	0.995	0.856
Beware of children	1884	106	1	0.994	0.995	0.884
CYCLE ROUTE AHEAD WARNING	1884	54	0.996	1	0.995	0.865
Dangerous Left Curve Ahead	1884	42	0.996	0.952	0.993	0.855
Dangerous Rright Curve Ahead	1884	72	0.972	1	0.986	0.822
End of all speed and passing limits	1884	48	0.998	1	0.995	0.824
Give Way	1884	107	1	0.985	0.995	0.834
Go Straight or Turn Right	1884	79	1	0.989	0.995	0.861
Go straight or turn left	1884	42	0.992	1	0.995	0.862
Keep-Left	1884	64	0.995	1	0.995	0.844
Keep-Right	1884	82	1	0.979	0.995	0.817
Left Zig Zag Traffic	1884	64	0.99	0.984	0.995	0.853
No Entry	1884	83	1	1	0.995	0.795
Overtaking by trucks is prohibited	1884	47	0.997	1	0.995	0.822
Pedestrian Crossing	1884	47	0.992	1	0.995	0.873
Round-About	1884	67	0.998	1	0.995	0.876
Slippery Road Ahead	1884	101	1	0.981	0.995	0.91
Speed Limit 20 KMPh	1884	48	0.996	0.771	0.968	0.77
Speed Limit 30 KMPh	1884	75	1	0	0	0
Stop_Sign	1884	32	0.985	1	0.995	0.881
Straight Ahead Only	1884	63	0.998	1	0.995	0.865
Traffic_signal	1884	6	0.723	0.667	0.633	0.289
Truck traffic is prohibited	1884	83	0.995	1	0.995	0.843
Turn left ahead	1884	84	0.996	0.976	0.993	0.844
Turn right ahead	1884	80	0.975	0.986	0.988	0.858
Uneven Road	1884	77	0.998	1	0.995	0.887

FIGURE 4.21 – Le résultat de train .

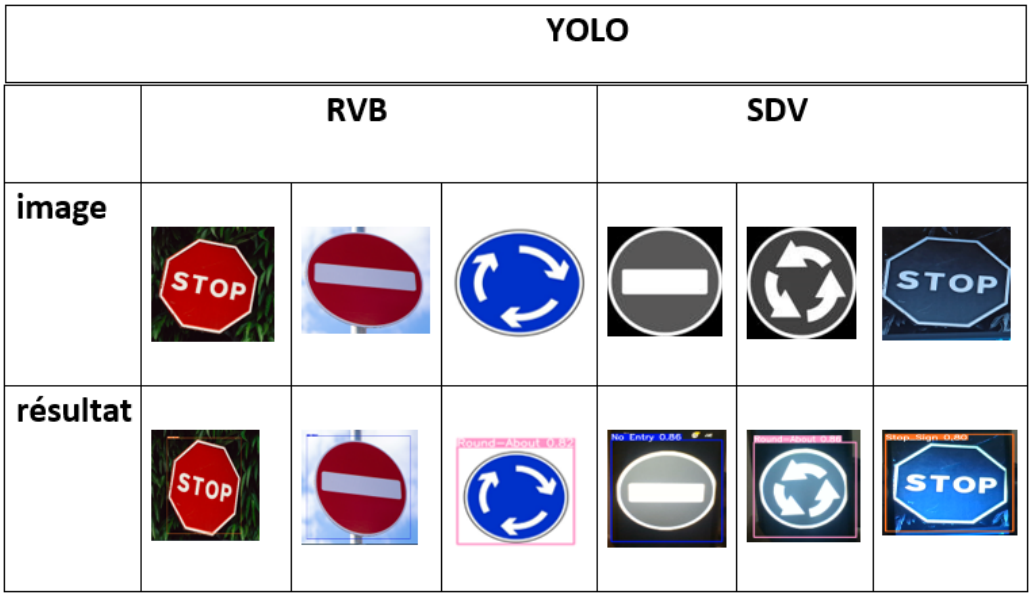

FIGURE 4.22 – Le résultat de programme .

basé sur data set rgb et ont obtenu des résultats de 0.95, tandis que dans notre projet, nous avons travaillé sur dataset sdv, ce qui réduit la taille des images et améliore leur qualité, ce qui aide à les apprendre rapidement, et nous avons atteint un bon résultat de 0.93.

4.5 Conclusion

Ce chapitre présente le modèle de segmentation sémantique YOLO que nous avons utilisé pour prédire les zones de panneaux de signalisation dans les images et leur type. Les résultats obtenus sont acceptables et démontrent l'efficacité de notre approche.

CONCLUSION GÉNÉRALE :

Les feux de signalisation jouent un rôle important dans la régulation de la circulation routière en fournissant des informations cruciales aux conducteurs. Toutefois, il arrive parfois que les conducteurs ignorent les feux de signalisation, en particulier en raison de la fatigue, des conditions météorologiques défavorables ou lorsqu'ils cherchent une adresse. Dans certains cas, les conducteurs dépassent également la limite de vitesse, ce qui non seulement augmente le risque d'accidents, mais également le risque de blessures graves et de décès en cas d'accident.

La détection d'objets dans les images, notamment la reconnaissance de panneaux de signalisation, est un domaine de recherche en vision artificielle qui trouve des applications dans divers domaines, notamment la vidéosurveillance, la sécurité routière et les systèmes d'assistance à la conduite. Nous avons particulièrement porté notre attention sur l'algorithme YOLO pour la découverte de formes.

Notre travail consistait à développer un détecteur de panneaux de signalisation sur data set sdv en utilisant l'algorithme YOLO et à l'intégrer dans un système de détection de panneaux routiers. Nous avons testé notre application sur différentes images et les résultats obtenus ont été satisfaisants. Nous avons arrivé à un taux de reconnaissance de 93% sur l'ensemble des données sdv dataset de 10000 images. Dans le futur nous essayons de travailler sur l'ensemble de tous les données existant

BIBLIOGRAPHIE

[1] Z. Chen, J. Yang, and B. Kong, "A robust traffic sign recognition system for intelligent vehicles," in *2011 Sixth International Conference on Image and Graphics*. IEEE, 2011, pp. 975–980.

[2] B. Soheilian, A. Arlicot, and N. Paparoditis, "Extraction de panneaux de signalisation routière dans des images couleurs," in *Reconnaissance des Formes et Intelligence Artificielle*, 2010, pp. 1–8.

[3] B. S. Shekar and G. Harish, "A machine learning model for detection and recognition of traffic signs," in *2021 International Conference on Intelligent Technologies (CONIT)*. IEEE, 2021, pp. 1–4.

[4] J. Torresen, J. W. Bakke, and L. Sekanina, "Efficient recognition of speed limit signs," in *Proceedings. The 7th International IEEE Conference on Intelligent Transportation Systems (IEEE Cat. No. 04TH8749)*. IEEE, 2004, pp. 652–656.

[5] M. Benallal and J. Meunier, "Real-time color segmentation of road signs," in *CCECE 2003-Canadian Conference on Electrical and Computer Engineering. Toward a Caring and Humane Technology (Cat. No. 03CH37436)*, vol. 3. IEEE, 2003, pp. 1823–1826.

[6] N. Barnes and A. Zelinsky, "Real-time radial symmetry for speed sign detection," in *IEEE Intelligent Vehicles Symposium, 2004*. IEEE, 2004, pp. 566–571.

[7] G. Loy and N. Barnes, "Fast shape-based road sign detection for a driver assistance system," in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566)*, vol. 1. IEEE, 2004, pp. 70–75.

[8] H. H. Aghdam, E. J. Heravi, and D. Puig, "A practical approach for detection and classification of traffic signs using convolutional neural networks," *Robotics and autonomous systems*, vol. 84, pp. 97–112, 2016.

[9] R. Qian, Y. Yue, F. Coenen, and B. Zhang, "Traffic sign recognition with convolutional neural network based on max pooling positions," in *2016 12th International conference on natural computation, fuzzy systems and knowledge discovery (ICNC-FSKD)*. IEEE, 2016, pp. 578–582.

[10] Y. Yang, S. Liu, W. Ma, Q. Wang, and Z. Liu, "Efficient traffic-sign recognition with scale-aware cnn," *arXiv preprint arXiv :1805.12289*, 2018.

[11] D. Cireşan, U. Meier, J. Masci, and J. Schmidhuber, "Multi-column deep neural network for traffic sign classification," *Neural networks*, vol. 32, pp. 333–338, 2012.

[12] P. Sermanet and Y. LeCun, "Traffic sign recognition with multi-scale convolutional networks," in *The 2011 international joint conference on neural networks*. IEEE, 2011, pp. 2809–2813.

[13] J. Jin, K. Fu, and C. Zhang, "Traffic sign recognition with hinge loss trained convolutional neural networks," *IEEE transactions on intelligent transportation systems*, vol. 15, no. 5, pp. 1991–2000, 2014.

[14] [Online]. Available : https://www.capterra.fr/glossary/201/advanced-driver-assistance-systems-adass

[15] [Online]. Available : https://larevuetech.fr/systeme-avance-aide-conduite-adas/

[16] [Online]. Available : http://www.lerepairedesmotards.com/dossiers/niveaux-conduite-autonome-automatisation-vehicules.php

[17] [Online]. Available : https://www.roulez-lesprit-libre.com/blog-auto/technologies-automobiles/voiture-autonome/

[18] [Online]. Available : https://www.spiceworks.com/tech/iot/articles/what-is-adas/

[19] [Online]. Available : https://www.normequip.com/487-panneau-de-signalisation

[20] [Online]. Available : https://www.lepermislibre.fr/code-route/cours/panneaux-code-route-typologie-signification

[21] [Online]. Available : https://www.intensive-driving-school.co.uk/traffic-lights-explained?fbclid=IwAR170Nx0MJ-QOH-Uw88HP7u8wtm8781rSHYMwSj9SwKp1w3IGczZOrC7EIk

[22] P.-A. Brousseau, "Calibrage de caméra fisheye et estimation de la profondeur pour la navigation autonome," 2020.

[23] [Online]. Available : https://www.techno-science.net/glossaire-definition/Methode-de-Viola-et-Jones.html

[24] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, vol. 1. Ieee, 2005, pp. 886–893.

[25] [Online]. Available : https://towardsdatascience.com/r-cnn-fast-r-cnn-faster-r-cnn-yolo-object-detection-algorithms-36d53571365e

[26] [Online]. Available : https://iq.opengenus.org/single-shot-detector/

[27] S. Zhang, L. Wen, Z. Lei, and S. Z. Li, "Refinedet++ : Single-shot refinement neural network for object detection," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 2, pp. 674–687, 2020.

[28] T. Kim, Y.-W. Tai, and S.-E. Yoon, "Pca based computation of illumination-invariant space for road detection," in *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2017, pp. 632–640.

[29] X. Xu, T. Yu, X. Hu, W. W. Ng, and P.-A. Heng, "Salmnet : A structure-aware lane marking detection network," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 8, pp. 4986–4997, 2020.

[30] M. Sozzi, S. Cantalamessa, A. Cogato, A. Kayad, and F. Marinello, "Automatic bunch detection in white grape varieties using yolov3, yolov4, and yolov5 deep learning algorithms," *Agronomy*, vol. 12, no. 2, p. 319, 2022.

[31] Z. Liu, M. Qi, C. Shen, Y. Fang, and X. Zhao, "Cascade saccade machine learning network with hierarchical classes for traffic sign detection," *Sustainable Cities and Society*, vol. 67, p. 102700, 2021.

[32] E. Güney, C. Bayilmiş, and B. Cakan, "An implementation of real-time traffic signs and road objects detection based on mobile gpu platforms," *IEEE Access*, vol. 10, pp. 86 191–86 203, 2022.

[33] M. ZERROUGUI and S. HAMADENE, "Détection de la tumeur cérébrales dans l'image irm par l'apprentissage en profondeur." Ph.D. dissertation, Université Mohamed el-Bachir el-Ibrahimi Bordj Bou Arréridj Faculté de ..., 2021.

[34] D. Learning, "Ian goodfellow, yoshua bengio, aaron courville," *The reference book for deep learning models*, vol. 1, 2016.

[35] [Online]. Available : https://www.simplilearn.com/tutorials/deep-learning-tutorial/deep-learning-algorithm

[36] J. Redmon and A. Farhadi, "Yolo9000 : better, faster, stronger," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7263–7271.

[37] ——, "Yolov3 : An incremental improvement," *arXiv preprint arXiv :1804.02767*, 2018.

[38] F. MESBAH, "Détection d'objets par deep neural network à l'aide du modèle yolo en temps réel." 2021.

[39] M. A. B. Zuraimi and F. H. K. Zaman, "Vehicle detection and tracking using yolo and deepsort," in *2021 IEEE 11th IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE)*. IEEE, 2021, pp. 23–29.

[40] S. S. Guia, A. Laouid, R. Euler, M. A. Yagoub, A. Bounceur, and M. Hammoudeh, "A salient object detection technique based on color divergence," in *The 4th International Conference on Future Networks and Distributed Systems (ICFNDS)*, 2020, pp. 1–6.

[41] Y. Derfoufi, "Programmation en langage python," 2019.

[42] [Online]. Available : AboutOpenCVâĂİ.In:().url:https://opencv.org/about

[43] [Online]. Available : AboutTensorflowâĂİ.In:().url:https://www.tensorflow.org/

[44] [Online]. Available : https://keras.io/.DatedudernieraccÃĭs:21/06/2021

[45] [Online]. Available : https://datascientest.com/matplotlib-tout-savoir

[46] [Online]. Available : https://www.journaldunet.fr/web-tech/guide-de-l-intelligence-artificielle/1501871-pytorch140922/

[47] [Online]. Available : https://pandas.pydata.org/

[48] [Online]. Available : https://datascientest.com/numpy/

[49] [Online]. Available : https://research.google.com/colaboratory/faq.html/

[50] [Online]. Available : :https://venturebeat.com/business/3-ways-to-overcome-the-cybersecurity-skills-gap/

[51] [Online]. Available : https://www.blogdumoderateur.com/tools/visual-studio-code/

[52] [Online]. Available : https://github.com/joshwadd/Deep-traffic-sign-classification