

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



UNIVERSITÉ Echahid Hamma Lakhdar - EL-OUED

FACULTÉ DE SCIENCES EXACTES

DEPARTEMENT D'INFORMATIQUE



MASTERACADEMIQUE

Domaine: Mathématiques et Informatique

Filière: Informatique

Spécialité: Systèmes Distribués et Intelligence Artificiel

Présenté par:

Melle Gueniaa fatma Zohra

Melle Tourqui ikram

Thème

**Comparaison de méthodes de classification
appliquées à la détection d'objets**

Devant le jury composé de:

Mr. **KHEBBACHE Mohib eddine**

Mr. **kERTHIOU Ismail**

Mr. **HAMOUD Meriem**

MAA Univ. ElOued Rapporteur

MAA Univ. El Oued Président

MAA Univ. ElOued examinateur

Année universitaire 2015/2016

Remerciements

Nous remercions Allah le tout puissant, qui nous a donné la force et la patience pour l'accomplissement de ce travail.

Nous adressons nos vifs remerciements à notre encadreur Mr. KHEBBACHE MOHIB EDDINE, qui nous a aidés pendant toute la durée de notre travail et pour sa patience et ses précieux conseils et ses compétences.

Et pour avoir accepté de nous encadrer.

Notre remerciements s'adressent également à notre enseignant Mr. ZAIZ FAOUZI qui nous a aidés beaucoup malgré son charge de travail.

Nous remercions vivement les membres du jury qui nous ont fait l'honneur d'accepter de juger notre travail.

N'oublions pas, bien sûr, nos professeurs de tout le cycle universitaire, de leur présenter nos meilleurs vœux

Et enfin, nous tiens à remercier tous ceux qui ont contribué, de près ou de loin, par leurs encouragements, leurs conseils et leurs soutiens à mener à bien ce travail.



Dédicaces

Avant tout, je remercie ALLAH le tout puissant de m'avoir donné le courage et la patience pour réaliser ce travail malgré toutes les difficultés rencontrées.

Tous d'abord Je dédie ce modeste travail

À ma très cher mère AOUADI HANJAA qui m'a encouragé par tous les moyens à avoir la confiance et l'espoir au cours

de la rédaction de ce travail. adorée et qui m'a aidé, grâce à leur prière et à leur bénédiction.

À mon cher père

Je le dédie aussi À ma très cher sœur KOUTHER ,SABRIN,ET SALMA.

A mes très chers frères MOHAMED, AMINE.

À ma très cher cousine NABILA. Et à toute la famille.

A mon très cher fiancé GUEMAZ SLIMANE

À mes meilleurs amies TRABLES, SALWA, SALMA, AZIZA, ELATRA ,NAJOTA

À ma binôme FATMA .

À tous les étudiants de la faculté Informatique surtout les étudiants de la 2ème année master promotion 2016

À tous les habitants de TAGHZOUT



TOURQUI IKRAM



Dédicaces

*Avant tout, je remercie **ALLAH** le tout puissant de m'avoir
donné le*

*courage et la patience pour réaliser ce travail malgré toutes les
.difficultés rencontrées*

Tous d'abord Je dédie ce modeste travail

*À mon très cher père **GUENIAA ABD ELKADDER** qui m'a
encouragé par tous les moyens à avoir la confiance et
l'espoir au cours*

de la rédaction de ce travail.

*À ma très cher mère adorée qui m'a aidé, grâce à leur prière et
à leur bénédiction.*

*Je le dédie aussi À ma très cher sœur **HALIMA**.*

*A mes très chers frères **BILAL** , **FODIL** et **YOUSEF**.*

*À ma très cher cousine **ZAHIA**. Et à toute la famille.*

*À mes meilleurs amies **SAFA**, **HADDA**, **IKRAM***

.À Tous mes amies d'étude

À tous mes amies des cités universitaire 1000 lit nouveau

*À tous les étudiants de la faculté Informatique surtout les étudiants
de la 2ème année master promotion 2016*

*.À tous les habitants de **EL_MEGHAIR***



GUENIAA FATMA ZOËRA

Résumé

La vision par ordinateur est un vaste domaine de recherche dont lequel un ordinateur est doté de capacité de percevoir, analyser et comprendre le monde qu'il observe d'une manière similaire au système visuel de l'être humain. Les méthodes de ce domaine sont généralement utilisées dans de nombreuses applications telles que la vidéosurveillance, l'aide à la conduite ou l'imagerie médical. Ces différentes applications s'appuient généralement sur la détection ou le suivi qui s'articule sur la problématique de représentation, qui utilise un descripteur pour l'extraction des caractéristiques d'objets dans les images, et celle de reconnaissance (classification) d'objet, qui repose sur un classificateur (algorithme de machine Learning) afin de distinguer les classes d'objets d'intérêt. Dans ce mémoire, nous adressons la problématique du choix d'un classifieur convenable qui met un défi face aux recherches dans ce domaine. Notre objectif visé est de comparer expérimentalement les performances des classifieurs multiclassés (SVM, KNN) à différencier les catégories d'objets dans un environnement routier. Pour comparer leurs performances, nous utilisons l'implémentation sous MATLAB de ces deux méthode de classification, ainsi, nous utilisons le descripteur (SURF) et aussi notre base d'images collectée depuis Internet afin d'entraîner et de tester ces classifieurs. Le taux de bonne reconnaissance est utilisé comme une métrique pour analyser leur performance.

Mots clés: vision par ordinateur, détection d'objets reconnaissance de d'objets, classification, apprentissage automatique, MATLAB.

Abstract

Computer vision is a vast field of research in which a computer has the ability to collect, analyze and understand the observed world with a similar manner as the human visual system. In this area, the methods are generally used in many applications such as video surveillance, driver assistance or medical imaging. These applications typically rely on detecting or tracking which focused on the representation issues, that used a descriptor for feature extraction, and the object recognition (classification), based on classifier (Learning machine algorithm) to distinguish classes of objects of interest. In this thesis, we address the issue of selection of a suitable classifier that faces the research in this area. Our objective is to compare experimentally the performance of multiclass classifiers (SVM, KNN) to categorizing objects in a road environment. To compare their performance, we use the MATLAB implementation of these two classification method and we use the descriptor (SURF) and also our dataset collected from the Internet to train and test these classifiers. The rate of correct recognition is used as a metric to analyze their performance.

Keywords: computer vision, object detection, object recognition, classification, machine learning, MATLAB.

المخلص

الرؤية الحاسوبية (أو عن طريق الحاسوب) هو ميدان بحث واسع بحيث يعمل فيه على تزويد الكمبيوتر بالقدرة على جمع وتحليل وفهم العالم المحيط به بطريقة مماثلة للنظام البصري للإنسان. وتستخدم أساليب هذا المجال من البحث عموماً في العديد من التطبيقات مثل المراقبة بالفيديو، مساعدة السائق أو التصوير الطبي. تعتمد هذه التطبيقات عادة على الكشف (detection) أو التتبع (suvi) التي تركز على إشكالية تمثيل الكائنات، وذلك لاستخراج أوصاف الأشياء في الصور، وكذلك إشكالية التعرف (التصنيف) على الكائنات التي تعتمد على المصنف (خوارزمية التعلم الآلي machine Learning) للتمييز بين فئات الأشياء المثيرة للاهتمام. في هذه الأطروحة، نريد أن نعالج إشكالية اختيار المصنف المناسب والتي تمثل تحدياً للبحث في هذا المجال. هدفنا هو المقارنة بالتجربة على أداء المصنفات المتعددة الفئات SVM ، KNN للتمييز بين أنواع الكائنات الممكنة مصادفتها على الطرق. لاجل هاته الدراسة المقارنة للأداء، ونحن نستخدم الأكواد الخاصة بالمصنفين السابقين والمتوفرة على تطبيق MATLAB مع استخدام الوصف (SURF) وأيضاً قاعدة بيانات للصور التي تم جمعها من الإنترنت لتدريب واختبار هذه المصنفات. يتم استخدام معدل التعرف الصحيح كمقياس لتحليل أدائها.

كلمات البحث: رؤية الكمبيوتر، والتعرف على وجوه الكشف عن وجوه، تصنيف، تعلم الآلة، Matlab.

Résumé.....	I
Glossaire.....	II
Table de matières.....	III
Liste des figures.....	IV
Liste des tableaux.....	VI
Introduction générale	VII

Table des matières

Chapitre01 : Vision par ordinateur et détection d'objet

Introduction.....	5
I. intelligence artificielle	5
I.1 De l'intelligence	6
I.2 Al' intelligence artificielle:	7
II. vision.....	7
II.1 vision humaine	8
II.2 vision par ordinateur.....	8
III. Reconnaissance de forme.....	
III.1 reconnaissance de forme	
III.2 apperntissage automatique	9
III.2.1 type d'apperntissage.....	9
III.3 reconnaissance d'objet.....	
IV. detection d'objet dans les images	10
IV .1. Processus de detectiond'objet.....	11
IV.2 type de capteur utilise pour la detction:	11
V. Defis et enjeux.....	
VI. Domaine d'application.....	18
Conclusion	19

Chapitre02 : Etat de l'Art da classification

Introduction.....	23
I. Description d'image	24
I.1 Descripteur local	25
I.2 Descripteur global.....	26
II. Apprentissage et classification.....	
II.1 Base d'appretissage	28
II.2 Classification	28
II.2.1 Classifieur.....	29
II.2.2 Methode de classification.....	30
II.2.2.1 Taxonomie de méthodes.....	30
II.2.2.2 Methode.....	32
II.3 Cascade de classifieurs	34
II.4 Défis de l'apprentissage du classifieur	35

III. Approches de détection d'objets : Etat de l'art	35
III.1 Bilan	37
Conclusion :	37

Chapitre 03 : Analyse comparative des performances

Introduction.....	37
II. Métriques de performance d'un classifieur.....	38
II.1 Classifieur binaire.....	38
II.2 Classification multiclassés :	39
III. Etude expérimentale des performances de classifieurs.....	39.
III.1. Description de l'environnement MATLAB	40
III.2 Base d'images(dataset).....	40
III.3 Evaluation de performance.....	41
III.4 Comparaison des résultats et discussions.....	41
Conclusion :	42
Conclusion générale	45

Liste des figures

Chapitre01 : Vision par ordinateur et détection d'objet

Figure 1.1 : reconnaissance d'instances d'objet (objets spécifiques) [16]	4
Figure 1.2 : reconnaissance de catégorie d'objet (objet générique) [15].....	5
Figure 1.3 : Le principe de fonctionnement d'un détecteur. [15].....	7
Figure 1.4 processus de détection en ligne d'objets [13].....	11
Figure 1.5 : les différents défis liés à la reconnaissance d'objets[16].....	13
Figure 1.6 : applications de vision par ordinateur optique de caractères(OCR [16].....	14
Figure 1.7 : applications de vision par ordinateur Contrôle de la machine [16].....	15
Figure 1.8 : applications de vision par ordinateur Imagerie médicale [16].....	19
Figure 1.9 : applications de vision par ordinateur surveillance et le trafic de surveillance [16].	20
Figure 1.10 : applications de vision par ordinateur Sécurité routière [15].....	21

Chapitre02 : état de l'Art de la classification

Figure 2.1 : Détermination de l'orientation principale d'un point d'intérêt SURF[2]	24
Figure 2.2 : Méthode à noyau SVM. [10]	25
Figure 2.3 : l'architecture d'un classifieur en cascade [2].	25

Chapitre 03 : Analyse comparative des performances

Figure 3.1 : Taux reconnaissance pour chaque classe de SVM	24
Figure 3.2 : Taux de reconnaissance pour classifieur knn	25
Figure 3.3 : Taux globale pour chaque classifieur.	25

Liste de tableau

Chapitre02 : état de l'Art de la classification

Tableau2.1 : les noyaux les plus utilisés ainsi que leurs paramètres pour SVM [10].

Tableau2.2 : tableau récapitulatif de différents travaux de recherche

Chapitre 03 : Analyse comparative des performances

Tableau3.1 : Matrice de confusion binaire [2]

Tableau3.2 : Matrice de confusion multi-classes [38]

Tableau3.3: Matrice de confusion de la classification multi classe pour le noyau linéaire

Tableau3.4 :Matrice de confusion de la classification multi classe pour le noyau RBF ($\sigma = 2$)

Tableau3.5: Matrice de confusion de la classification multi classe le noyau RBF ($\sigma = 4$)

Tableau 3.6: Matrice de confusion de la classification multi classe pour le noyau Polynomial ($p=4$)

Tableau 3.7: Matrice de confusion de la classification multi classe pour le noyau Polynomial ($p=2$)

Tableau 3.8: Matrice de confusion de la classification multi classe pour knn($K=1$)

Tableau 3.9: Matrice de confusion de la classification multi classe pour KNN($K=3$)

Tableau 3.10: Résumé des résultats pour la classification de notre système avec la méthode SVM

Tableau 3.11: Résumé des résultats pour la classification multi classe

Tableau 3.12: Résumé des résultats pour la classification multiclassées avec KNN et SVM

Glossaire

IA : Intelligence Artificielle

OCR : Optical Character Recognition

ADAS : Advanced Driver Assistance System

SIFT: Scale Invariant Feature Transform

SURF: Speeded Up Robust Features

HOG : Histogram of Oriented Gradients

SVM: Support Vector Machine

KNN: k-Nearest Neighbors

RVM : Relevance Vector Machine

ROC: Receiver Operating Characteristics

CVPR : Computer Vision and Pattern Recognition

OVA : One-versus-all

AVA : All-versus-all

VP : Vrais positifs

FP : Faux positifs

FN : Faux négatifs

VN : Vrais négatifs

TGBR : Taux global de bonne reconnaissance

KPP : K Plus Proche Voisin

MATLAB : Matrix Laboratory

Introduction général

Chez l'être humain, la reconnaissance visuelle des scènes et des objets, via le système de vision humaine, est généralement rapide et automatique, d'où un être humain peut détecter la présence des milliers d'objets instantanément et sans effort. De nos jours, les travaux de recherche autour l'intelligence artificielle tentent d'imiter les ordinateurs avec les capacités sensoriels et de décision afin de simuler le système visuel de l'être humain, on parle de domaine de vision par ordinateur.

Dans le contexte de la vision par ordinateur, la détection d'objets est un processus cognitif important dans la perception, la compréhension des scènes et des objets et aussi dans la prise de décision. Ce dernier est appliqué dans plusieurs domaines de recherche et notamment dans les Véhicules Intelligents. Dans cette optique, les véhicules sont dotés des systèmes intelligents embarqués d'aide à la conduite. Le rôle de ces systèmes est d'assister activement le conducteur dans sa prise de décision en basant sur la détection d'objets autour de l'environnement proche du véhicule comme les bords de route, les obstacles routiers. Ceci dans le but d'éviter l'accident et assurer la sécurité routière.

Cependant, pour détecter l'existence un objet dans une image, il est indispensable de prendre en compte les apparences et les formes que peuvent avoir les. Cela revient à caractériser l'objet (en utilisant un descripteur) par un ensemble de caractéristiques de forme, de texture ou bien d'apparence dans le but de le représenter au mieux pour et le distinguer des autres objets. Après, un processus de classification, repose sur un classifieur, consiste à reconnaître un objet en comparant ses caractéristiques avec celles des catégories connues des objets (les caractéristiques extraites d'un objet à partir d'exemples étiquetés), afin de prédire la classe (catégorie) à laquelle l'objet d'intérêt pourrait se rattacher (classification binaire) ou les classes d'objets (classification multi classes). En effet, la problématique de la détection d'objets se résume autour de deux problèmes majeurs: la représentation et la classification.

La problématique que nous procurons, s'articule autour du choix de la méthode de classification la plus performante dans certaines conditions.

Notre objectif, dans ce mémoire, est de réaliser une étude expérimentale pour comparer les performances de certains classifieurs multiclassés afin de prédire (détecter) de façon hors ligne la présence d'objets de certaines catégories dans une scène routière.

Afin qu'on puisse atteindre cet objectif, nous avons choisi trois catégories d'objets présents dans des scènes routières: piétons, véhicule et animaux. Ainsi, notre étude est basée sur un processus d'apprentissage hors ligne implémenté sous MATLAB version 2015b en basant sur notre propre base d'images d'apprentissage et de test (dataset) collectée depuis Internet. Tout d'abord, nous utilisons le descripteur d'images (SURF) pour extraire des informations des images. Puis, à partir de ces informations, deux classifieurs sont entraînés à différencier les objets d'intérêt des autres objets. Pour la classification, nous avons recours à des méthodes multiclassées à noyaux (SVM, KNN). Chaque méthode a été paramétrée, testée et validée sur notre base d'images. L'exactitude de détection (le taux de bonne reconnaissance) est utilisée comme une métrique de performance pour la comparaison.

Nous avons organisé ce présent mémoire en trois chapitres:

- **Chapitre I:** dans ce chapitre nous avons présenté des concepts de bases liés à vision par ordinateur et détection d'objets (reconnaissance de forme, apprentissage automatique, détection en ligne...).
- **Chapitre II:** ce chapitre met l'accent sur les méthodes de classification ainsi que particulièrement les fonctionnalités principales d'un classifieur. à la fin de ce chapitre nous avons cités quelques travaux dans ce domaine suivis par une synthèse.
- **Chapitre III:** ce dernier chapitre faisant principalement l'objet d'évaluation de performances. Tout d'abord nous avons présenté les métriques d'évaluation ainsi que le choix des méthodes et paramètres. Après nous avons conclu ce chapitre par une discussion de résultats obtenus.

Finalement, nous clôturons le travail faisant l'objet de ce mémoire par une conclusion générale, qui fournit aussi quelques perspectives d'amélioration au futur.

Chapiter 01

Vision par ordinateur et détection d'objet

1. Introduction

Dans l'ère moderne, les systèmes de vision sont radicalement répartis et répandus dans presque tous les domaines de la vie. Pour simuler les systèmes de vision réel, les caméras sont devenues installées dans tous les lieux et connectées à tous les détails de nos vies, tout en ouvrant la porte vers d'autres domaines basés sur les caméras, comme capteurs, tels que la vision par machine et la vision par ordinateur.

A ce stade, la détection d'objets est fortement trouvée dans certaines applications telles que la vidéosurveillance (dans les magasins, rues ou aéroports), les systèmes d'aide à la conduite (aide au guidage ou détection d'obstacles), et bien d'autres applications. Cependant, malgré les avancées de la vision par ordinateur, les systèmes développés restent loin d'être performants par rapport aux systèmes de vision humaine basés sur l'œil et du cerveau humain.

Ce premier chapitre est une introduction à notre sujet qui est traité dans certains travaux de recherche et notamment dans le domaine de vision par ordinateur. En fait, nous allons présenter un panorama sur les différents concepts de bases qui sont liés à ce domaine de recherche tels que la reconnaissance de formes, l'apprentissage, les types d'apprentissages, ...etc.

2. Intelligence Artificielle

2.1 De l'intelligence

Dans la littérature, la notion d'intelligence a été définie selon plusieurs points de vue. Dans notre contexte, cette notion peut être définie comme la capacité de prendre la bonne décision, selon un certain critère. Pour prendre de meilleures décisions, des connaissances sont nécessaires, et elles doivent être dans un format opérationnel : elles doivent pouvoir être utilisées pour interpréter des données sensorielles et prendre des bonnes décisions en conséquence.[1]

2.2 A l'Intelligence artificielle

Le terme d'Intelligence Artificielle (IA) désigne toutes les techniques et méthodes qui permettent de doter des systèmes informatiques de capacités proches de celles de l'être humain. Les systèmes développés ne sont capables que d'imiter certaines capacités de l'homme. Dans cette optique, les robots dotés d'une intelligence via des capacités de perception de l'environnement et de commande sont des exemples pratiques. [2]

De nos jours, les domaines et les applications de l'IA sont diverses et variées. Parmi eux on trouve une branche consacrée à la détection et reconnaissance basée sur la vision. Cette dernière s'articule sur des méthodes qui, à partir de l'observation visuelle d'un objet (photo, séquence vidéo, image médicale, image

satellite, etc.) provenant des capteurs « entrées sensoriels », prennent une décision afin de prédire la classe attribuée à cet objet. Par exemple, un système qui, à partir d'une observation radar, classe des objets volants soit dans une classe avions, soit dans une classe non-avions, réalise une tâche de reconnaissance d'objets. [2]

3 vision

3.1 vision humaine

La vision humaine est la perception visuelle des rayonnements lumineux observés par l'œil, c'est-à-dire, l'interprétation cognitive du sens de la vue, partagé par de nombreuses espèces animales.

La vision recouvre l'ensemble des processus physiologiques et psychologiques par lesquels la lumière émise ou réfléchiée par l'environnement détermine les détails des représentations sensorielles, comme les formes, les couleurs, les textures, le mouvement, la distance et le relief. Ces

processus cognitifs complexes font intervenir l'œil, organe récepteur de la vue, mais aussi des zones spécialisées du cerveau (voir cortex visuel).[3]

3.2 Vision par ordinateur

La vision par ordinateur consiste à tenter d'imiter la vision humaine ou animale (ex : vision à larges champs de certains oiseaux, de certains insectes) par des composants électroniques [4]. En d'autres termes, elle tente de doter l'ordinateur par des capacités de perception de l'environnement semblable au système visuel d'un être humain ou d'une espèce d'animal.

Dans autre sens « La vision par ordinateur » ou « La reconnaissance visuelle par ordinateur » consiste à prédire à l'aide d'un algorithme quel objet se trouve dans une image. En raisonnant en termes d'entrées et sorties d'un algorithme, l'entrée de l'algorithme est une image, la sortie est la classe de l'objet (tâche de reconnaissance de catégorie d'objets) ou la référence d'un objet précis (tâche de reconnaissance d'instance d'objet). Si l'image contient plusieurs objets, la sortie de l'algorithme peut être une liste d'objets. Cette tâche, appelée « reconnaissance d'objets » diffère de la localisation d'objet ou de l'appariement d'objet. La première consiste à déterminer précisément la position d'un objet dans une image, la seconde consiste à trouver les correspondances point à point entre deux objets présents dans deux images.

4.Reconnaissance d'objets

Tout d'abord, avant de mettre l'accent sur le domaine de recherche moderne de la reconnaissance d'objets en vision par ordinateur, nous mettrons un coup d'œil sur les deux aspects fondamentaux de ce domaine-ci, à savoir, la reconnaissance de formes et l'apprentissage automatique.

4.1 Reconnaissance de formes

La reconnaissance de formes (ou *pattern recognition* en anglais) est un thème récurrent dans le domaine de l'intelligence artificielle. Le but de la reconnaissance de formes consiste à analyser un ensemble de données en comparant leurs caractéristiques avec des configurations ou des modèles connus à l'aide des méthodes de classification. Les applications de la reconnaissance de formes sont nombreuses, et certaines méthodes sont d'ores et déjà utilisées au quotidien, notamment pour la reconnaissance d'écriture et de la parole, la traduction, ou encore les prévisions météorologiques. Avec l'évolution des capacités des machines et l'amélioration des méthodes de reconnaissance de formes, de nouvelles applications peuvent ainsi être envisagées[16], comme, par exemple, la détection d'obstacle (piéton, véhicule, animal...) dans une scène routière.

En outre, le processus de reconnaissance de formes peut être accompli sur plusieurs étapes successives [16]:

- **Acquisition** : assimilation des données, prétraitements.
- **Représentation** : extraction de caractéristiques pertinentes,
- **Analyse** : discrimination de la représentation.
- **Post-traitements** : validation de la décision.

4.2 Apprentissage automatique

L'apprentissage automatique fait référence au développement, à l'analyse et à l'implémentation de méthodes qui permettent à une machine d'évoluer grâce à un processus d'apprentissage, et ainsi de remplir des tâches qu'il est difficile ou impossible de remplir par des moyens algorithmiques plus classiques ,l'objective de apprentissage c'est un extraire et exploiter automatiquement l'information présenter.

En formalisme mathématique le plus fréquemment utilisé pour décrire l'apprentissage. Nous disposons d'exemples d'entraînement $\mathcal{D} = \{z_1, z_2, \dots, z_n\}$ où les z_i sont des exemples générés par un processus **inconnu** $P(Z)$.

Nous avons également une fonction d'erreur L , qui prend comme arguments une fonction de décision f et un exemple z , et qui renvoie un scalaire réel. Nous voulons minimiser l'espérance de $L(f, Z)$ sous le processus stochastique $P(Z)$.

4.2.1 Types d'apprentissage

Les méthodes d'apprentissage peuvent être classées en plusieurs catégories, à savoir : apprentissage supervisé, apprentissage semi-supervisé, apprentissage non-supervisé et apprentissage par renforcement .

4.2.1.1 Apprentissage supervisé

L'apprentissage supervisé a pour but d'apprendre par l'exemple. Il faut fournir au préalable une liste d'objets avec leurs étiquettes de classe - appelée ensemble d'apprentissage - afin que le système soit capable d'expliquer et ensuite de prédire l'appartenance d'un nouvel objet à une classe connue a priori. Beaucoup d'algorithmes d'apprentissage supervisé sont utilisés pour faire de la reconnaissance d'objets : caractères, visages, personnes...[2] .

4.2.1.2 Apprentissage semi-supervisé

Ce terme regroupe des méthodes qui se situent entre l'apprentissage non-supervisé et l'apprentissage supervisé. Ce type de méthodes est utilisé quand un grand nombre de données est disponible mais sans qu'elles soient toutes étiquetées. L'initialisation de la méthode est faite à partir d'un petit jeu de données correctement étiquetées. Puis l'algorithme doit lui-même étiqueter les exemples suivants et construire son propre modèle.

Les algorithmes d'apprentissage non-supervisé et semi-supervisé sont beaucoup utilisés pour la recherche d'informations sur internet notamment. Ils permettent de traiter ainsi une grande quantité de données.

4.2.1.3 Apprentissage non-supervisé

Pour ce type d'apprentissage, l'ensemble d'apprentissage est seulement composé d'exemples, sans aucune étiquette de classe. C'est à l'algorithme de trouver des dépendances, des structures entre les différents exemples. Le « clustering » ou partitionnement de données regroupe un ensemble de méthodes d'apprentissage non-supervisé, comme l'algorithme des K-means [6] ou l'Isodata [7]. Les classes (ou « clusters » en anglais) sont créées par l'algorithme qui regroupe dans une même classe des objets ayant des caractéristiques communes entre elles et différentes avec les objets n'appartenant pas aux mêmes classes.[2]

4.2.1.4 Apprentissage par renforcement

Ce type de méthode est un apprentissage interactif. A chaque décision que l'algorithme prend, il reçoit en retour des réponses de l'environnement appelées signaux de renforcement. C'est un processus adaptatif qui améliore la solution en fonction des réponses qu'il reçoit. L'algorithme de « Q-learning » [5] est un algorithme type un apprentissage par renforcement. Les algorithmes par renforcement permettent d'établir des processus plus complexes, comme le guidage de robots. Ce dernier a un objectif final à atteindre et en fonction des réponses de ses différents capteurs, il va pouvoir affiner ses actions.

4.3 Reconnaissance d'objet

Dans le domaine de la vision humaine, un adulte peut reconnaître en général plus de 10 000 catégories d'objets et une quantité bien plus importante d'instances d'objets. Ce processus de reconnaissance est souvent rapide, sans effort, et robuste aux changements de point de vue, de luminosité, à des occultations d'une partie de l'objet. Cependant, en vision par ordinateur, il y a un grand intérêt à programmer des ordinateurs pour qu'ils soient eux aussi capables de reconnaître des catégories d'objets ou des instances d'objets de la même manière [8].

L'approche générale consiste à considérer que l'on dispose d'une banque de données où sont stockés des modèles d'objets et une vue du monde réel. Puis on vérifie l'existence de l'objet extrait dans la banque de données, donc on parle du problème de catégorisation. Pratiquement, cette modélisation peut être géométrique [8].

Dans la majorité des cas, une phase d'apprentissage préalable, où quelques images de la catégorie ou de l'instance d'objet sont fournies, est nécessaire afin de construire des modèles de prévision par le biais des méthodes de classification supervisée. A partir de ce modèle l'ordinateur prédit si une image contient ou non cette catégorie d'objet (reconnaissance de catégorie d'objet) ou cet objet spécifique (reconnaissance d'instance d'objet) [8].

Généralement, dans la reconnaissance d'objet il existe deux types différents de tâches :

- **la reconnaissance des instances d'objets:** ce type, associé aux Choses et est également connu comme la reconnaissance d'objet spécifique, vise à identifier les instances d'objets précédemment vu comme objet particulier, lieu, une personne ou une voiture, et est en grande partie un problème d'adaptation, dans lequel les différences entre les exemplaires mémorisés et les objets à reidentifiés dans une image d'entrée sont principalement causées par les changements de condition de formation d'image. Toutefois, l'apprentissage d'une nouvelle instance d'objet peut se faire à partir d'une seule

image de cet objet [7]. Dans la figure , le visage de Carl Gauss, la Tour Eiffel, ou un certain magasin sont des exemples d'instances d'objets [15].



Figure1.1: reconnaissance d'instances d'objet (objets spécifiques) [15].

- **la reconnaissance des classes (catégories) d'objets:** ce type associé aux Formes, est également connu comme la reconnaissance d'objet générique, met l'accent sur la reconnaissance de certaines catégories prédéfinies, et dans la pratique, il faut non seulement de déterminer si toutes les instances de catégories d'intérêt ou non sont présents dans une image d'entrée, mais aussi de les localiser avec précision dans l'image pour les séparer de l'arrière-plan. L'apprentissage d'une nouvelle catégorie d'objets peut se faire à partir d'un petit nombre d'images de cet objet [15]. La figure montre des exemples d'objets de catégories bâtiments et voitures [15].



Figure1.2: reconnaissance de catégorie d'objet (objet générique) [15]

5. Détection d'objets dans les images

En plus d'être essentiel à la reconnaissance des classes d'objets dans les scènes réelles, la détection d'objets est également une étape pré-requise dans un large éventail de tâches de vision de plus haut niveau, y compris la reconnaissance d'événements ou d'activités, la compréhension du contenu d'une scène, etc.

A cet effet, la détection d'objets est devenue l'un des secteurs les plus ciblés dans la vision par ordinateur tout en motivant les travaux de recherche autour ce domaine au cours de ces dernières décennies. [11]

Pratiquement, la détection d'objet en vision par ordinateur est traitée comme un problème de classification d'objet pour prédire la présence d'une instance ou d'une classe (catégorie d'intérêt) d'objets dans une image numérique [11]. Ces méthodes font souvent appel à l'apprentissage supervisé et ont des applications dans de multiples domaines, tels la recherche d'image par le contenu ou la vidéo surveillance, la détection de visage et la détection de personne, etc.[9]

Pour fonctionner, le détecteur (système de détection) fait appel à une phase d'apprentissage hors ligne. Dans cette phase, le classifieur est entraîné et enseigné sur des données correspondant aux objets à reconnaître (les différentes classes) par le biais de base d'apprentissage comprenant déjà des images « prédécoupées » autour d'objets ciblés. Ceci permet de distinguer la classe d'intérêt (par exemple, les voitures) de toute autre chose en utilisant ces fonctionnalités.[2]

A l'issue de cette phase, la phase de traitement (détection et reconnaissance) en ligne est alors possible[2]. Après avoir décidé la représentation (description) d'objets et d'entraîner le classifieur, on peut alors utiliser la valeur de décision de ce classifieur pour déterminer la présence /absence d'objets dans un nouvelle image.[15]la figure illustre ce principe de fonctionnement

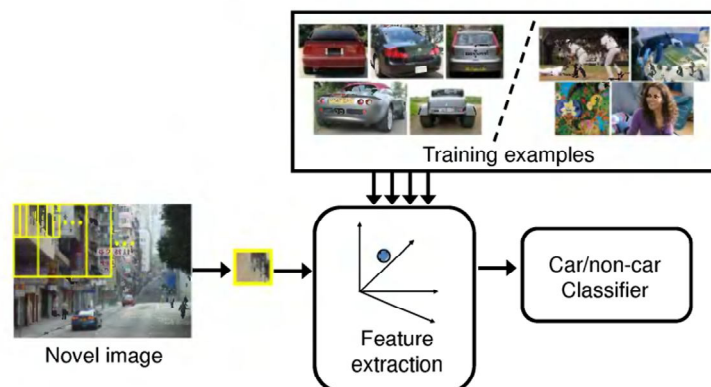


Figure1.3: Le principe de fonctionnement d'un détecteur. [15]

En outre, en vue qu'un objet dans une scène du monde réel occupe souvent juste une (petite) partie de la scène et est occulté par d'autres objets ou structures similaires de l'arrière-plan, la détection d'objets ne vise seulement à reconnaître et déterminer la classe (catégorie) d'objets qui sont présents dans une image d'entrée, mais aussi de localiser avec précision les instances de catégories d'intérêt dans cette image pour les séparer de l'arrière-plan [11].

5.1 Processus de détection d'objets

Dans cette section, nous allons maintenant décrire brièvement les étapes de ce processus.



Figure1.4 : processus de détection en ligne d'objets [13]

Comme montre la figure , la détection en ligne d'objets dans une scène du monde réel est décrit par les étapes suivantes :

- **Acquisition de l'image**: La capture et la perception de la scène de monde réel est la première étape dans le processus. Il faut réussir à capter l'information pertinente sans bruit par le biais des capteurs (Un dispositif physique qui est sensible à une bande du spectre d'énergie électromagnétique). Une conversation numérique (digitalisation) est utilisé pour convertir l'image acquit sous forme signal électrique (images analogiques) en une forme [14].L'image dans cette étape est dans un état brut ce qui engendre un risque de bruit qui peut dégrader les performances du système

- **Phase de prétraitement** : Le prétraitement consiste à préparer les données issues du capteur à la phase suivante. Il s'agit essentiellement de réduire le bruit superposé aux données et essayer de ne garder que l'information significative de la forme représentée. Le bruit peut être dû aux conditions d'acquisition (éclairage, mise incorrecte du document, ...). Parmi les opérations de prétraitement généralement utilisées on peut citer : l'extraction des composantes connexes, le lissage, la normalisation et la squelettisation. [16]

• **Phase de segmentation** : lors de la détection et reconnaissance en ligne, les données proviennent d'un capteur basé sur la vision couvre une large scène et l'objet recherché n'en occupe qu'une partie. Afin de pouvoir détecter puis reconnaître cet objet, il est nécessaire de trouver des régions qui représentent des objets d'intérêt ou des parties significatives d'objets avant tout traitement de niveau supérieur [2]. Pour cela, et à partir d'une image acquise, la segmentation divise l'image en régions (segments) connexes, homogènes et différentes de ses voisins correspondant à des objets. Les segments «significatives» peuvent être utilisés pour extraire des informations utiles et décrire le contenu d'une image (analyse d'image), comme un segment de "fond (background)", segments pour "objets", ou catégories d'objets en particulier []. Généralement, la segmentation d'une image est effectuée par l'utilisation de l'une des deux grandes approches, à savoir, basée sur l'extraction de contours (frontières) ou la croissance des régions. [2]

- **L'approche contour** : L'approche contour consiste à identifier les transitions entre les régions. L'approche contour n'aboutit pas directement à une segmentation, car les contours détectés ne sont pas toujours connexes. Il existe cependant des techniques permettant d'obtenir des contours fermés. Dans ce cas, on observe une parfaite dualité entre les contours et les régions Dans cette approche la détection de contours consiste à balayer l'image avec une fenêtre définissant la zone d'intérêt. A chaque position, un opérateur est appliqué sur les pixels de la fenêtre afin d'estimer s'il y a une transition significative au niveau de l'attribut choisi. Littérature est très riche par ces techniques. [2]
- **L'approche région** : Les méthodes de l'approche région cherchent à regrouper directement des pixels ayant une propriété commune ; l'ensemble des regroupements de pixels définit à la fin une segmentation de l'image. Les plus importantes sont celles qui procèdent par croissance de régions et par division/fusion de régions : La segmentation par croissance de régions vise à regrouper les pixels adjacents de l'image dont les attributs varient de façon négligeable. Il faut choisir un prédicat ou critère d'uniformité P et un jeu de pixels initiaux, ou germes, à partir desquels la croissance de régions s'effectuera. Les méthodes du type division et fusion comportent, comme leur nom l'indique, deux étapes. Dans la première, l'image est divisée récursivement jusqu'à ce que toutes les régions vérifient le prédicat P. Dans la deuxième étape, des régions adjacentes sont regroupées, tant que les régions résultantes vérifient P. [2]
- **Approche Coopération** : Concernant la segmentation, il est acquis depuis une dizaine d'années déjà qu'il faut faire coopérer les approches régions et les approches contours un bon critère de segmentation. Indépendamment des problèmes inverses à régulariser, de nombreux travaux ont été développés ces dernières années sur des modèles de contours. Ces modèles de contours sont

développés pour la segmentation d'objet dans une image et n'étaient pas, jusqu'à récemment encore, envisagés comme des modèles de contours pour la segmentation par approche contour/région. Les travaux actuels tentent de combiner et d'unifier ces différents modèles.

- **Extraction des caractéristiques (description):** avant qu'un classifieur entraîné sur un ensemble d'apprentissage puisse déterminer de l'existence d'un objet dans une image, il est crucial de prendre en compte les apparences et les formes que peuvent avoir les objets. En fait, cette étape consiste à résoudre le problème de représentation d'objet. Pour cela un ensemble d'informations caractérisant l'objet doivent être extraites à partir d'un modèle de représentation. D'une façon générale, l'extraction des caractéristiques consiste à extraire des descripteurs de forme, de texture ou bien d'apparence dans le but de représenter au mieux l'image en fonction de la tâche à réaliser.[10]

- **Phase de Classification :** cette phase est primordiale et est connu comme une phase d'analyse d'image où en utilisant un classifieur entraîné pendant l'apprentissage, le système peut indiquer si l'objet recherché est présent ou non dans chaque image. A ce stade l'objectif est d'identifier la classe des objets génériques dans une image.[10] En imagerie routier par exemple, on cherche à déterminer des objets de catégories piétons, véhicules, panneaux routières, animaux (ex : chameaux). Dans ce domaine et à partir des caractéristiques extraites dans la phase précédente, le classifieur attribue les objets de l'image à des classes connues à priori (c'est la classification supervisée) ou à des classes inconnues (classification non supervisée).

- **Phase post-traitement :** Cette dernière phase permet d'exploiter le résultat obtenu par la phase d'analyse. Cette étape permet ainsi de valider la décision ou de la rejeter. Il est possible de tenir compte d'informations supplémentaires qui pourront renforcer le choix de l'acceptation du résultat. Le contexte permet également de valider la décision. Des informations extérieures et différentes des données acquises peuvent être utilisées pour ajouter une information pertinente pour aider à la décision finale. Il est également possible de définir des contraintes sur la confiance liée à la décision. Si celle-ci est trop faible, a décision finale sera alors repoussée ultérieurement si d'autres observations peuvent étayer le résultat[2].

5.2 Types de capteur utilisés pour la détection d'objet

Généralement, deux types de capteurs qui sont, à savoir : les capteurs actifs et passifs [10].

a) Détection par des capteurs actifs : Un capteur est dit "actif" s'il produit une énergie envoyée sur la scène ; les obstacles sont détectés par l'énergie réfléchiée par leur surface. Un capteur actif possède donc à la fois un émetteur et un récepteur. La télémétrie laser et le radar représentent les deux capteurs actifs les

plus utilisés sur des véhicules ou sur des robots d'extérieur, parce qu'ils sont robustes aux variations des conditions atmosphériques (pluie, neige) ou à des conditions dégradées d'illumination.

▪ **Exemples sur les capteurs actifs :**

✚ **télémètre Laser:** La télémétrie laser combine un système électronique pour la mesure ponctuelle de distance à partir d'un faisceau laser réfléchi par l'obstacle, et un mécanisme de déviation du faisceau émis et de balayage pour acquérir des coupes ou des images de distance dans une région d'intérêt quelconque. La lumière réfléchie par un obstacle quelconque se trouvant dans l'axe du tir revient vers le capteur et est perçue par un récepteur; la distance capteur-obstacle est obtenue de manière classique par le temps de vol d'une impulsion laser, ou par le déphasage d'un signal modulé en amplitude ou en fréquence. Tous les impacts laser sont dans un plan si le balayage est seulement en azimut, ou sont dans l'espace 3D si le balayage est en site et azimut.

L'emploi de la télémétrie laser est très courant sur des véhicules haut de gamme, ou sur des robots de service, car la mesure est rapide, la portée peut être importante (plusieurs centaines de mètres selon le type de télémètres utilisé et les taux d'échecs (faux négatifs ou faux positifs) très faibles ,et les télémètres les plus répandus sont:

- ✓ télémètre laser IBEO pour détecter les objets dans la scène et retourne une image de profondeur haute résolution.
- ✓ Capteur LMS200 de Sick Inc. en Allemagne : ce capteur est fréquemment utilisé sur les robots pour la navigation à vitesse modérée.
- ✓ Capteur Velodyne .

✚ **Radar :** Le radar (RADioDetection And Ranging) consiste à mesurer les ondes radio réfléchies par les objets métalliques, en exploitant la théorie de Doppler pour détecter leurs déplacements en fréquence. Le radar peut être exploité pour détecter des objets avec de très grandes portées ; il peut être utilisé en statique (par exemple, détection d'intrusion dans un environnement sensible) ou mobile (par exemple, détection d'obstacles depuis un véhicule).

b) Détection d'objets par capteurs passifs basées sur la vision : Un capteur passif ne fait que recevoir l'énergie naturellement réfléchi par les objets présents dans la scène. Plusieurs capteurs passifs sont utilisés pour la robotique ou pour la surveillance :

- ✓ les capteurs inertiels (détection de vibrations, de mouvement).
- ✓ les capteurs haptiques (contact).
- ✓ les capteurs visuels qui sont les plus connus, donc des caméras CCD ou CMOS. En tant que capteur passif, une caméra mesure l'intensité lumineuse émise par l'environnement sous la forme d'une image digitalisée, ou d'un tableau de pixels.

▪ Exemples sur les capteurs passifs

- ✚ **La stéréo-vision:** Les systèmes de stéréovision permettent de calculer par triangulation, la profondeur des objets dans l'espace 3D, grâce à leurs projections dans les plans des deux caméras. La différence des positions dans les deux images, de la projection d'un même point de l'espace 3D, s'appelle la disparité ; la recherche des points correspondants entre ces deux images, permet donc de produire une image de disparité. En supposant que les caméras ont des axes optiques parallèles, alors la disparité est inversement proportionnelle à la profondeur. Ainsi la disparité est nulle pour un point à l'infini, par exemple un point sur la ligne d'horizon.
- ✚ **Capteur de vision Infrarouge (IR) :** Malgré leur faible résolution et le mauvais contraste des images qu'elles fournissent, les caméras infrarouges sont exploitées pour la navigation de véhicules dans des conditions de visibilité difficiles, soit la nuit, soit en cas de mauvaises conditions météo par exemple avec du brouillard ou de la neige. On trouve dans la littérature de nombreux travaux sur la détection d'objets depuis des caméras IR de nuit ou avec de la fumée.

5.2 Défis et enjeux

La détection de classe d'objet impose un certain nombre de contraintes et enjeux qui peuvent face les techniques de reconnaissance et de classification. Dans cette section nous traitons ces derniers en les classifiant comme suit:

- **Défis liés à la robustesse :** dans la même catégorie (classe) d'objets, les différentes instances ont toujours différentes formes, poses, couleurs et textures. Ceci se réfère aux variations d'apparence intra-classes. Du même, en raison de variation de l'éclairage, fond, posture, et point de vue ainsi que l'influence des occultations, la construction d'un modèle d'apparence catégorique qui peut accueillir toutes les variations possibles intra-classe est très difficile.
- **Défis liés à l'apprentissage du classifieur :** Les propriétés qui sont propres à la tâche d'apprentissage dans la détection d'objets font des contraintes autour l'apprentissage de classifieur. Nous résumons ces défis en trois aspects : [11]

a) **défi de l'échelle :** La détection de la classe d'objets est une tâche avec une grande complexité. En outre, pour détecter un objet sous de nombreuses variantes telles que la pose, l'éclairage, la déformation, etc., l'ensemble des exemples positifs et négatifs d'entraînement (base d'apprentissage) doit être assez grand, allant des dizaines de milliers à plusieurs

milliards d'exemples. En fait, le nombre d'exemples nécessaires pour une tâche d'apprentissage, et ainsi la taille de la base d'apprentissage, augmente avec la complexité du problème ce qui influence négativement sur la performance de classifieur. [11]

b) défi de la vitesse : Une phase d'entraînement qui prend un temps long peut influencer sur la vitesse et le temps d'essai qui sont plus importantes pour la détection d'objet. Cependant, pour aboutir à une vitesse de détection d'objet plus rapide et en temps réel, il est nécessaire de réduire considérablement le temps d'entraînement. [11]

c) Défi d'asymétrie : la détection de l'objet comprend une tâche d'apprentissage fortement asymétrique. La propriété d'asymétrie contribue grandement à la difficulté de l'apprentissage du classificateur. [11]

• **Défis liés à la reconnaissance d'objets :** La difficulté majeure du problème de la reconnaissance d'objets repose sur les variations d'un objet ou des objets d'une même catégorie sur des images différentes. On peut grouper les sources de variations dans sept catégories principales : [11]

– **Les variations de point de vue :** les objets réels sont des entités physiques tridimensionnelles. Leur apparence change de façon significative en fonction du point de vue.

– **Les variations d'illumination :** en fonction de changements de l'environnement et des conditions d'acquisition d'une image, un objet peut subir des variations d'illumination importantes. Les objets dans les images des scènes naturelles sont très sensibles aux variations d'illumination qui modifient considérablement les couleurs et les valeurs radiométriques des objets.

– **Les occultations :** dans une image les objets peuvent être partiellement masqués par d'autres objets ou par leurs propres composants.

– **Les variations d'échelle :** les objets d'une même catégorie peuvent avoir des tailles physiques différentes ou les images d'un même objet peuvent avoir des tailles différentes en fonction de la résolution de l'image.

– **Les déformations :** la plupart des objets réels n'ont pas une forme rigide. Particulièrement les objets articulés subissent généralement des déformations linéaires et non-linéaires de leurs composants.

- **Complexité du fond** : en général, dans une image les objets ne sont pas isolés de leur environnement. Sur un fond complexe il est difficile de distinguer les régions appartenant à l'objet et au fond.
- **Variations intra-classe** : c'est l'un des problèmes importants envisagés pour la reconnaissance d'objets générique. Les objets d'une même classe peuvent avoir des apparences très variables, même quand on se limite aux catégories à un niveau élémentaire. Dans le cas extrême, les catégories fonctionnelles comme la classe « voiture » est définies par un haut niveau d'abstraction et les indices visuels seuls ne sont pas suffisants pour les reconnaître.

La figure montre les différents défis liés à la reconnaissance d'objets

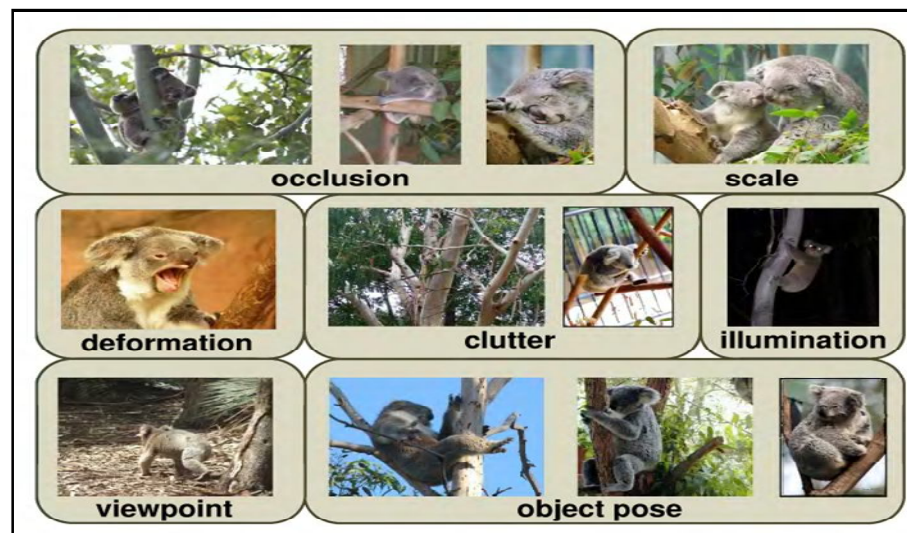


Figure1.5 : les différents défis liés à la reconnaissance d'objets[15]

6. Domaines d'applications

La vision par ordinateur est utilisée aujourd'hui dans une grande variété d'applications de monde réel, qui comprennent: [15]

- **Reconnaissance optique de caractères (OCR)**: la lecture de codes postaux manuscrits sur les lettres (Figure1.6) et la reconnaissance de plaque d'immatriculation automatique (ANPR) sont des exemples typiques. [15]

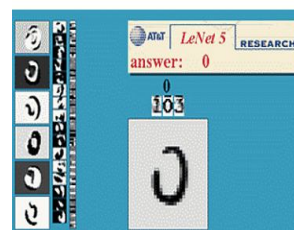


Figure1.6 : applications de vision par ordinateur reconnaissance optique de caractères(OCR [16]

- **Contrôle de la machine:** l'inspection rapide des parties pour l'assurance de qualité en utilisant la vision stéréo avec éclairage spécialisé pour mesurer les tolérances sur les ailes de l'avion ou des pièces de carrosserie (voir figure) ou à la recherche de défauts de pièces moulées en acier à l'aide de la vision à rayons X. [16]



Figure1.6 : applications de vision par ordinateur Contrôle de la machine [16]

- **Imagerie médicale:** l'enregistrement des images pré-opératoires et intra-opératoire ou d'effectuer des études à long terme de la morphologie du cerveau des gens à mesure qu'ils vieillissent. (voir figure)



Figure1.8: applications de vision par ordinateur Imagerie médicale [16]

- **Surveillance:** surveillance des intrus, l'analyse du trafic routier (Figure), et pools de surveillance pour les victimes de noyade.

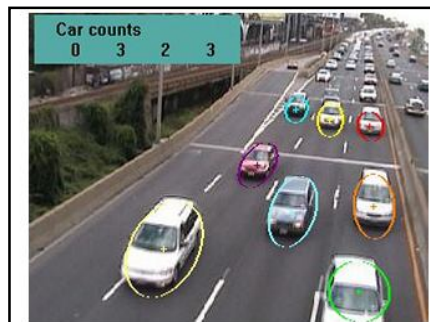


Figure1.9 : applications de vision par ordinateur la surveillance et le trafic de surveillance [16]

- Sécurité routière:** la détection des obstacles inattendus tels que les piétons sur la rue, dans des conditions où les techniques de vision actifs tels que le radar ou lidar ne fonctionnent pas bien est l'une des applications innovantes de domaine de vision par ordinateur afin d'assurer la sécurité au niveau de la route. Parmi les solutions envisagées on trouve les systèmes intelligents d'aide à la conduite (ADAS pour Advanced Driver Assistance System) [16]. Dans un système d'ADAS basé sur des caméras, les deux composantes principales sont le système de vision et le système de contrôle. Le système de vision est responsable du traitement des données brutes provenant des caméras et fournit des informations ou des alertes en temps réel à plus haut niveau au système de contrôle. Ce dernier alerte le conducteur et dans certains cas spécifiques doit ralentir ou stopper le véhicule. Ce type d'assistance peut s'avérer d'une grande utilité pour la détection des panneaux routiers, détection et suivi des piétons, les avertissements de changement de voie et les systèmes d'alerte anticollision. [16] La figure montre les différents systèmes ADAS ainsi que les capteurs embarqués dans un véhicule intelligent.

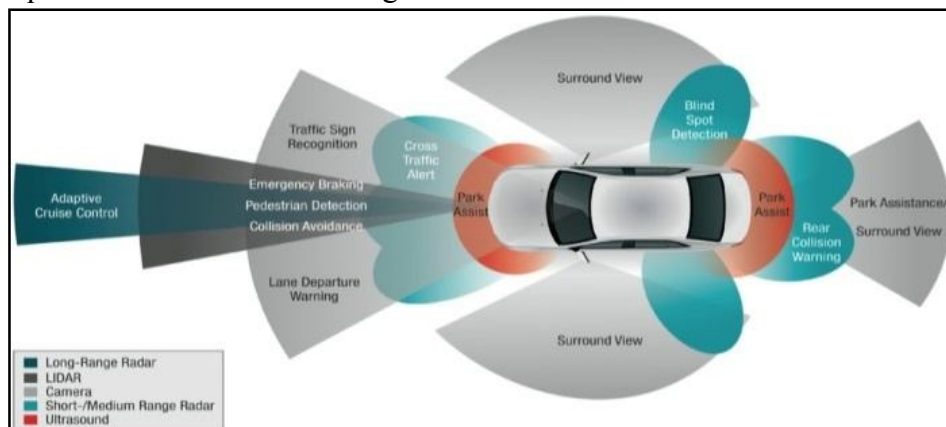


Figure1.9 : applications de vision par ordinateur Sécurité routière [15]

7. Conclusion

Dans l'intelligence artificielle, le domaine de vision par ordinateur et détection d'objets est un domaine de recherche très actif qui prend une place importante et nécessite la maîtrise de nombreux concepts de base.

Dans ce chapitre, nous avons présenté un panorama de certains concepts cruciaux, tels que la reconnaissance et l'apprentissage tout en précisant leur relation avec ce domaine. A la fin de ce panorama, nous pouvons conclure que notre sujet de recherche peut être formulé comme un problème de classification.

Dans le chapitre suivant, nous allons effectuer une étude sur l'état de l'art de la classification.

Chapiter 02

État de l'art Classification

1. Introduction

La grande problématique de la détection d'objet et la reconnaissance de formes est la classification, qui est pour but d'identifier les classes auxquelles appartiennent des objets à partir de certains paramètres descriptifs. Pour fonctionner, le classifieur a besoin de vecteur de caractéristiques, comme données d'entrée, fourni par un descripteur. Ainsi il a besoin d'une base d'apprentissage sur la quelle il s'entraîné pour qu'il puisse prendre de décisions.

Dans ce chapitre, nous allons voir la définition de terme de classification et les différentes méthodes de classification et les concepts en relation qui sont utilisés pour la détection d'objets dans une image. Ainsi, nous allons voir le principe de fonctionnement de classifieur et finalement nous allons présenter un état de l'art sur les travaux connexes à la détection d'objets où nous mettons sur les méthodes de classification, attaché avec un bilan synthétisant les remarques et observations.

2. Description d'images

Lorsque des zones de l'image ont été définies et sélectionnées, il faut analyser l'information qu'elles contiennent et extraire des zones d'intérêt sur des informations relatives à ce qui est contenu dans l'image, pour cela il existe des descripteurs d'images qui permettent de caractériser l'information disponible. Pratiquement deux types de descripteurs à savoir, local et global, peuvent être envisagés. [2]

2.1 Descripteur local

Est une méthode permette de détecter des points, puis le descripteur caractérise chaque point par ses voisinages, c'est-à-dire l'image est décrite en des points particuliers et significatifs, ces points intéressants sont appelés les points d'intérêt.

Un descripteur local d'un point d'intérêt fournit une caractérisation locale sous la forme d'un vecteur d'attributs. En effet, il y a des algorithmes **SIFT** [17] et **SURF** [18] permettent de détecter les points d'intérêt et de construire des descripteurs robustes à plusieurs transformations .

- 1) **descripteur SIFT (Scale Invariant Feature Transform):** est un algorithme proposé par Lowe en 2004 pour résoudre la rotation d'image, mise à l'échelle, et la déformation affine, le changement de point de vue, le bruit, les changements d'éclairage, a également une forte robustesse.

Pour l'algorithme SIFT, les étapes de détection, se résument comme suite:

- Sélection des pics dans l'espace d'échelles
 - Localisation des points d'intérêts (keypoints)
 - Calcul de l'orientation
 - Description du point d'intérêt [38].
- 2) **Descripteur SURF (Speeded Up Robust Features)** : SURF est un algorithme de détection de caractéristique et un descripteur, présenté par des chercheurs de l'ETH Zurich et de la Katholieke Universiteit Leuven pour la première fois en 2006 puis dans une version révisée en 2008. Il est utilisé dans le domaine de vision par ordinateur, pour des tâches de détection d'objet ou de reconstruction 3D. SURF est partiellement inspiré par le descripteur SIFT, qu'il surpasse en rapidité et, selon ses auteurs, plus robuste pour différentes transformations d'images. SURF est fondé sur des sommes de réponses d'ondelettes de **Haar** 2D et utilise efficacement les images intégrales. En tant que caractéristique de base, SURF utilise une approximation d'ondelettes de **Haar** du détecteur de blob à base de déterminant hessien [39]

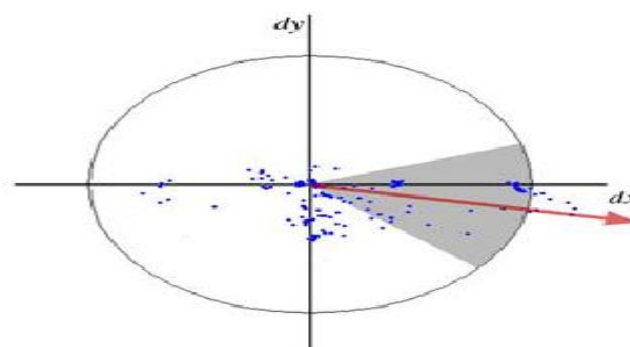


Figure2.1 : Détermination de l'orientation principale d'un point d'intérêt SURF[2]

2.2 Descripteur global

En reconnaissance d'objets, un descripteur global est plus facile à utiliser car il traite l'image dans son intégralité. Et tous les pixels de l'image correspondant à la zone d'intérêt sont pris en compte dans la description. Le descripteur est ainsi moins sensible aux déformations d'une image à l'autre. A l'heure actuelle, les descripteurs les plus utilisés en reconnaissance d'objets sont les ondelettes de **Haar** et les histogrammes de gradients orientés [19] du fait des résultats satisfaisants et rapides qu'ils permettent d'obtenir.

- 1) **les ondelettes**: La première définition de la Transformée en Ondelettes a été établie par Grossmann et Morlet en temps et échelle continus. La théorie des Ondelettes repose sur

l'idée que le signal peut être caractérisé par différentes échelles et différentes résolutions. En résumé, elle consiste à remplacer le paramètre de fréquence de la Transformée de Gabor par un paramètre d'échelle, ce qui conduit à une analyse temps-échelle et non plus temps-fréquence. Il existe plusieurs types d'ondelettes utilisées en imagerie comme l'ondelette de Gabor ou l'ondelette de Haar [2]...

- 2) **Histogrammes de gradients orientés:** Un histogramme de gradient orienté (HOG) est une caractéristique utilisée en vision par ordinateur pour la détection d'objet. La technique calcule des histogrammes locaux de l'orientation du gradient sur une grille dense, c'est-à-dire sur des zones régulièrement réparties sur l'image. Elle possède des points communs avec les SIFT, les Shape contexts et les histogrammes d'orientation de contours, mais s'en diffère notamment par l'utilisation d'une grille dense. La méthode s'est montrée particulièrement efficace pour la détection de personnes. Les HOG ont été proposés par Navneet Dalal et Bill Triggs, chercheurs à l'INRIA de Grenoble, à la conférence CVPR de juin 2005 [43].

3. Apprentissage et classification

Comme nous avons mentionné dans le premier chapitre, le détecteur (système de détection) fait appel à une phase d'apprentissage hors ligne, dans laquelle le classifieur est entraîné sur une base d'apprentissage comprenant des images correspondant aux objets à reconnaître. Avant d'entamer dans la présentation de différents concepts liés à la classification, nous faisons un clin d'œil sur les bases d'apprentissage.

3.1 Bases d'apprentissage

Les méthodes d'apprentissage peuvent être classées en plusieurs catégories : apprentissage supervisé, apprentissage semi-supervisé, apprentissage non-supervisé et apprentissage par renforcement (voir chapitre 1).

Pour l'apprentissage des modèles d'objets visuels et pour tester les performances de classification, le système a besoin d'une base de données d'images (dataset), appelée base d'apprentissage, afin d'entraîner le classifieur à distinguer des objets appartenant à différentes classes. Dans le cas de la classification entre les classes, il s'agit de différencier un type d'objet particulier par rapport à tout ce qui n'est pas cet objet. En effet, la base d'apprentissage contient un grand nombre d'exemples positifs et négatifs d'où les exemples positifs correspondent à la classe d'objets recherchés, tandis que les exemples négatifs correspondent à tout ce que n'est

pas cet objet. Le choix de la base d'apprentissage n'est pas trivial et est en rapport avec l'application visée[2]..

Dans la littérature, il y a eu plusieurs collections d'images accessibles au public, telles que : UIUC [20], Caltech 4 [21], et Caltech 101 [22], PASCAL VOC [23], et ImageNet [24]. Ces dernières ont joué un rôle clé dans la recherche de reconnaissance de catégorie d'objets et ainsi pour le développement et l'évaluation d'algorithmes de détection et suivi d'objets.[25]

Après avoir décidé la représentation (description) d'objets et d'entraîner le classifieur, on peut alors utiliser la valeur de décision de ce classifieur pour déterminer la présence /absence d'objets dans une nouvelle image. Par la suite nous décrivons les différents concepts liés à la classification ainsi que le principe de fonctionnement d'un classifieur.

3.2 Classification

La classification est généralement la dernière étape dans le développement d'un algorithme de vision par ordinateur, qui consiste à prendre les caractéristiques extraites de l'image et de les utiliser pour classer automatiquement les objets d'image. Cela se fait par le développement des algorithmes de classification qui utilisent les informations de fonctionnalité. Les mesures de distance ou de similarité sont utilisées pour comparer différents objets et leurs vecteurs de caractéristiques. [12].

D'une autre façon, on appelle classification, la catégorisation algorithmique d'objets. Elle consiste à attribuer une classe ou catégorie à chaque objet (ou individu) à classer, en se basant sur des données statistiques. Elle fait couramment appel aux méthodes d'apprentissage et est largement utilisée en reconnaissance de formes [12]..

L'algorithme de classification est communément appelé « classifieur » qui est généré lors de la phase de l'apprentissage hors ligne. Par la suite nous mettons l'accent sur ce concept.

3.2.1 Classifieur

Un classifieur désigne tout système de traitement de données qui reçoit une forme x et donne des informations à propos de la classe correspondant à cette forme. Plus particulièrement, il désigne un système qui génère des données pouvant se trouver dans un des N_w états possibles. Les données qui sont générées par ce système quand il se trouve dans un état i sont considérées comme appartenant à une classe w_i . [12]

Avant que le classifieur puisse fonctionner, il doit être s'entraîné sur l'ensemble des exemples d'apprentissage. I.e. durant la phase d'apprentissage hors ligne qui génère un

classifieur capable de différencier les objets d'une image. Cette étape doit permettre au classifieur de généraliser, c'est-à-dire d'attribuer une étiquette adéquate à un nouvel exemple dont seul le vecteur d'observation est connu. [26]

En outre, selon le nombre de classes à distinguer, le classifieur (ou la classification) est classé en deux catégories:

- **Classification binaire:** Classification binaire tomberait généralement dans le domaine de l'apprentissage supervisé depuis l'ensemble de données de formation est marqué. Et comme son nom l'indique, il est tout simplement un cas particulier dans lequel il n'y a que deux classes.
- **Classification multiclass:** Dans l'apprentissage de la machine, la classification multiclass ou multinomiale est le problème de la classification des cas dans l'un des plus de deux classes.

Tandis que certains algorithmes de classification permettent naturellement l'utilisation de plus de deux classes, d'autres algorithmes sont par nature binaires; ceux-ci peuvent, cependant, être transformés en classificateurs multinomiaux par une variété de stratégies. En outre, la classification multiclass ne doit pas être confondue avec la classification multi-étiquette, où plusieurs étiquettes sont à prédire pour chaque instance.

Pour réduire le problème de la classification multiclass, plusieurs procédés ont été proposés tels que:

- **un contre tous (One-versus-all (OVA)) :** L'approche la plus simple est de réduire le problème de la classification entre les K classes dans des problèmes K binaires, où chaque problème discrimine une classe donnée des autres K-1 classes. Pour cette approche, nous avons besoin de $N = K$ classifieurs binaires, où le k ème classifieur est formé avec des exemples positifs appartenant à la classe k et des exemples négatifs appartenant à l'autre $K - 1$ classes. Lors du test, par exemple, inconnu, le classifieur produisant l'output maximal est considéré comme le gagnant, et cette étiquette de classe est attribuée à cet exemple. Rifkin et Klautau [20] indiquent que cette approche, bien que simple, offre des performances qui est comparable à d'autres approches plus complexes lorsque le classificateur binaire est bien réglée.
- **Tous contre tous (All-versus-all (AVA)):** Dans cette approche, chaque classe est comparée à l'autre classe. Un classifieur binaire est conçu pour établir une discrimination entre chaque paire de classes, tout en rejetant le reste des classes. Ceci

requiert la construction de $K(K-1)/2$ classificateurs binaires. Lors de l'essai d'un nouvel exemple, un vote est effectué parmi les classificateurs et la classe avec le nombre maximum de votes gagne. Les résultats montrent que cette approche est en général mieux que l'approche d'un contre-tout.

3.2.2 Méthodes de classification

3.2.2.1 Taxonomie de méthodes

On peut grouper les méthodes classificatoires en deux grandes familles, à savoir : supervisée (Classement) et non supervisée (Classification) [1].

- 1) **Classification supervisé (classement)**: elle s'appuie sur l'apprentissage supervisé. En fait, les groupes sont fixés au préalable d'où la base d'image contient des exemples d'objets de chaque groupe. Le principe de la classification supervisée se base sur l'estimation d'une fonction de décision prédisant la classe d'une observation (une observation=vecteur de caractéristiques). Lorsque les distributions de probabilité des catégories ne sont pas connues dans l'espace de représentation, l'inférence des règles de décision est appelée apprentissage automatique de fonctions de décision. Lors de ce processus, l'algorithme apprend un modèle de classification à partir d'images labellisées, i.e. dont on connaît la classe. Ce modèle permet de prédire la classe d'une image non labellisée lors d'une phase de prédiction, appelée phase de test. Si l'algorithme n'a pas la capacité de généralisation sur un ensemble d'objets inconnus (jamais vus), ceci se traduit par le fait que l'algorithme commet beaucoup d'erreurs. cette situation pourrait être évitée en cherchant le minimum de l'erreur de généralisation sur un ensemble de validation.
- 2) **classification non supervisé**: on ne connaît pas de groupe. Cependant, Il existe d'autres types de classification qui s'appuient sur d'autres types de méthodes d'apprentissages comme « l'apprentissage semi-supervisé » et « l'apprentissage par renforcement ». En effet, l'apprentissage semi-supervisé est un bon compromis entre les deux types d'apprentissage « supervisé » et « non-supervisé », car il permet de traiter un grand nombre de données sans avoir besoin de toutes les étiqueter, et il profite des avantages des deux types mentionnés. Alors que l'apprentissage par renforcement est fortement utilisé dans le cas d'apprentissage interactif.

3.2.2.2 Méthodes

Les méthodes utilisées pour la classification sont nombreuses, nous citons:

- Séparateurs à Vastes Marges (SVM)
- KNN (k-nearest neighbor)
- Relevance Vector Machine (RVM)
- les Réseaux de Neurones
- Boosting

Dans cette partie nous allons détaillée les méthodes SVM et KNN utilisées dans le cadre de ce document.

a) méthode Séparateurs à Vaste Marge

Les machines à vecteurs supports (Support Vector Machines, SVM) sont des méthodes généralistes d'apprentissage et de discrimination. Ces méthodes ont montre leur efficacité dans de nombreuses applications. Les SVM ont été à l'origine des travaux de Vapnik [27] et ont été conçus pour la décision binaire. L'originalité principale de ces méthodes consiste à utiliser efficacement les exemples étiquètes afin de produire une fonction de décision qui maximise la marge entre deux classes données. L'efficacité repose sur la sélection des éléments d'apprentissage les plus représentatifs de la tâche de décision tout en maximisant la capacité de généralisation du modèle. Ces éléments correspondent aux exemples proches de la frontière de décision. Ils sont appelés *les supports vecteurs* et sont situes sur la marge. Pour garantir la bonne généralisation de la fonction de décision recherchée, le problème mathématique correspond à la maximisation de la distance des exemples annotes de la frontière (voir figure...).

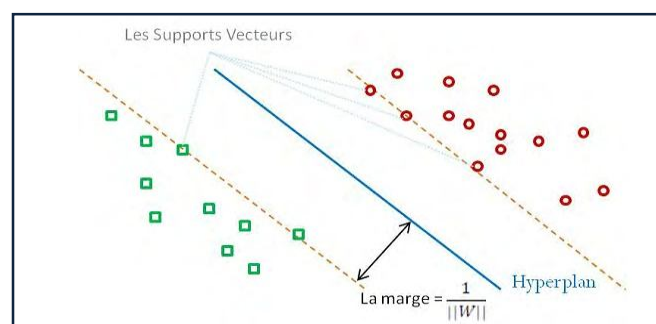


Figure2.2 : Méthode à noyau SVM. [10]

Cette méthode a montré son efficacité dans de nombreux domaines d'applications tels que le traitement d'image, la catégorisation de textes ou le diagnostics médicales et ce même sur

des ensembles de données de très grandes dimensions. Cependant, malgré qu'un algorithme de SVM soit en général peu coûteux en temps, il faut compter que la recherche des paramètres permettant d'obtenir la performance maximale, peut requérir des phases de test assez longues [5].

- **Les hyper paramètres de SVM [10] :**

La mise en œuvre de SVM requiert la détermination des valeurs d'hyper paramètres. Bien que ces paramètres soient inconnus a priori, ils sont décisifs pour obtenir un modèle performant de classification [3]. Les deux principaux paramètres à régler sont l'influence du noyau et de ses paramètres, et la complexité du modèle contrôlée par le paramètre de pénalité C . Un noyau permet de mesurer la similarité entre deux vecteurs caractéristiques x et x' . Le tableau.... illustre les deux noyaux les plus utilisés ainsi que leurs paramètres spécifiques.

Nom	Paramètre	Linéarité	Formule
Polynomial	Ordre : $p \geq 1$	Linéaire pour $p = 1$	$K(x, x') = (\langle x, x' \rangle + 1)^p$
Gaussien (RBF)	Largeur de bande : σ	Non-linéaire	$K(x, x') = \exp(-\frac{1}{2} \frac{\ x-x'\ ^2}{\sigma^2})$

Tableau2.1 : les noyaux les plus utilisés ainsi que leurs paramètres pour SVM [10].

En effet, les machines à noyaux peuvent donner a la fois les meilleurs et les moins bons résultats pour des valeurs différentes d'hyper paramètres. Donc, Le meilleur couple du paramètre est déterminé par validation. [10].

- **SVM pour la décision multiclassés [2]**

Les méthodes classiques d'utilisation des SVM pour le cas multiclassés consistent à décomposer, dans un premier temps, le problème en une série de dichotomies (un-contre-un, un-contre-tous). Ensuite, les décisions des classifieurs élémentaires sont combinées par une stratégie de fusion (vote, utilisation de la théorie probabiliste ou de l'évidence) pour permettre la discrimination multiclassés. L'approche « un-contre-un » propose d'utiliser $n(n-1)/2$ discriminateurs binaires pour décrire toutes les dichotomies possibles parmi les n classes. Quant à l'approche « un-contre-tous », elle utilise n classifieurs binaires dont chacun est spécialisé pour la reconnaissance d'une classe opposée à la fusion des $(n - 1)$ autres classes. Du point de vue performance de classification, aucune de ces approches n'est meilleure dans tous les cas. Toutefois, il convient d'utiliser l'approche un-contre-un afin de pouvoir associer des probabilités à chacun des SVM binaires. L'application d'une méthode de fusion à partir de ces probabilités semble donner de meilleurs résultats [HT98, Can07].

b) méthode des k plus proches voisins KNN

K-nn (K-nearest neighbor) [28] est une méthode d'apprentissage supervisée qui raisonne avec le principe sous-jacent : “dis moi qui sont tes amis, je te dirais qui tu es”. Elle diffère des méthodes d'apprentissages traditionnelles car aucun modèle n'est induit à partir d'exemple. A chaque fois que l'on veut classer un nouvel individu, on refait tourner l'algorithme et on cherche de nouveaux amis. KNN figure parmi les plus simples algorithmes d'apprentissage artificiel. Dans un contexte de classification d'une nouvelle observation x , l'idée fondatrice simple est de faire voter les plus proches voisins de cette observation. La classe de x est déterminée en fonction de la classe majoritaire parmi les k plus proches voisins de l'observation x . La méthode KNN est donc une méthode à base de voisinage, non-paramétrique ; Ceci signifiant que l'algorithme permet de faire une classification sans faire d'hypothèse sur la fonction $y=f(x_1, x_2, \dots, x_p)$ qui relie la variable dépendante aux variables indépendantes. [32].

- **Quelques règles sur le choix de k [32].**

Le paramètre k doit être déterminé par l'utilisateur : $k \in \mathbb{N}$. En classification binaire, il est utile de choisir k impair pour éviter les votes égalitaires. Le meilleur choix de k dépend du jeu de donnée. En général, les grandes valeurs de k réduisent l'effet du bruit sur la classification et donc le risque de sur apprentissage, mais rendent les frontières entre classes moins distinctes. Il convient donc de faire un choix de compromis entre la variabilité associée à une faible valeur de k contre un « oversmoothing » ou surlissage (i.e. gommage des détails) pour une forte valeur de k . Un bon k peut être sélectionné par diverses techniques heuristiques, par exemple, de validation-croisée. La valeur de k choisi doit assurer un minimum d'erreur de classification.

- **Algorithme 1-NN [32].**

La méthode du plus proche voisin est une méthode non paramétrique où une nouvelle observation est classée dans la classe d'appartenance de l'observation de l'échantillon d'apprentissage qui lui est la plus proche, au regard des covariables utilisées. La détermination de leur similarité est basée sur des mesures de distance. Formellement, soit L l'ensemble de données à disposition ou échantillon d'apprentissage : $L = \{(y_i, x_i), i = 1, \dots, nL\}$ où $y_i \in \{1, \dots, c\}$ dénote la classe de l'individu i et le vecteur $x(i) = (x_{i_1}, \dots, x_{i_p})$ représente les variables prédictives de l'individu i . La détermination du plus proche voisin est basée sur un fonction distance arbitraire $d(\cdot, \cdot)$. La distance euclidienne ou dissimilarité entre deux individus caractérisés par p covariables est définie par:

$$d((x_1, x_2, \dots, x_p), (u_1, u_2, \dots, u_p)) = \sqrt{(x_1 - u_1)^2 + (x_2 - u_2)^2 + \dots + (x_p - u_p)^2}$$

Ainsi, pour une nouvelle observation (y, \mathbf{x}) le plus proche voisin $(y(1), \mathbf{x}(1))$ dans l'échantillon d'apprentissage est déterminé par: $d(\mathbf{x}, \mathbf{x}(1)) = \min_i(d(\mathbf{x}, \mathbf{x}_i))$ et $\hat{y} = y_1$, la classe du plus proche voisin, est sélectionnée pour la prédiction de y_i . Les notations $\mathbf{x}(j)$ et $y(j)$ représentent

$$d(\mathbf{x}_i, \mathbf{x}_j) = \left(\sum_{s=1}^p (x_{is} - x_{js})^2 \right)^{\frac{1}{2}}$$

respectivement le $J^{\text{ème}}$ plus proche voisin de \mathbf{x} et sa classe d'appartenance. Parmi les fonctions distance types, la distance euclidienne est définie comme suit :

Et plus généralement la distance de **Minkowski** : $d(\mathbf{x}_i, \mathbf{x}_j) = \left(\sum_{s=1}^p |x_{is} - x_{js}|^q \right)^{\frac{1}{q}}$

- **Algorithme KNN [32].**

Une première extension de cette idée, qui est largement et communément utilisée en pratique, est la méthode des k plus proches voisins. La plus proche observation n'est plus la seule observation utilisée pour la classification. On utilise désormais les k plus proches observations. Ainsi la décision est en faveur de la classe majoritairement représentée par les k voisins. Soit k_r le nombre d'observations issues du groupe des plus proches voisins appartenant à la classe r .

$$\sum_{r=1}^c k_r = k$$

Ainsi une nouvelle observation est prédite dans la classe l avec : $l = \max_r (k_r)$
Ceci évite que la classe prédite ne soit déterminée seulement à partir d'une seule observation. Le degré de localité de cette technique est déterminé par le paramètre k : pour $k=1$, on utilise la méthode du seul plus proche voisin comme technique locale maximale, pour $k \rightarrow n-1$ on utilise la classe majoritaire sur l'ensemble intégral des observations (ceci impliquant une prédiction constante pour chaque nouvelle observation à classifier).

- **Les étapes de la classification KNN [32].**

1. Soit L un échantillon d'apprentissage constitué des observations \mathbf{x}_i relatives à une classe y_i :

$$L = \{(y_i, \mathbf{x}_i), i = 1, \dots, nL\},$$

Soit \mathbf{x} une nouvelle observation, dont la classe y doit être prédite :

$$\hat{y} = ?$$

2. sélection des $(k+1)$ plus proches voisins de \mathbf{x} selon une fonction distance $d(\cdot, \cdot)$ préalablement choisie : $d(\mathbf{x}, \mathbf{x}_i)$

3. Standardisation des k plus petites distances via le $(k+1)$ ème voisin :

$$D_{(i)} = D(\mathbf{x}, \mathbf{x}_{(i)}) = \frac{d(\mathbf{x}, \mathbf{x}_{(i)})}{d(\mathbf{x}, \mathbf{x}_{(k+1)})}$$

4. Transformation des distances normalisées $D(i)$ en poids $w(i)$ à partir d'une fonction noyau $K(\cdot)$: $w(i)=K(D(i))$

5. La classe de x est choisie d'après la majorité pondérée des k plus proches voisins :

$$\hat{y} = \max_r \left(\sum_{i=1}^k w_{(i)} I(y_{(i)} = r) \right)$$

3.3 Cascade de classifieurs [2]

Un problème complexe tel que la reconnaissance des objets nécessite un très grand nombre de caractéristiques conduisant à des temps de traitement très importants. Pour réduire la charge de calcul, [29] ont proposé de construire une cascade de classifieurs de complexité croissante sélectionnant un nombre faible de caractéristiques aux premiers étages. Le modèle de cascade de classifieurs est essentiellement utilisé pour la détection et la reconnaissance d'objets représentés par régions. L'approche par régions consiste à décomposer l'image d'un objet en nombreuses régions avant de caractériser chacune d'entre elles. Ensuite, des classifieurs peuvent être employés sur ces régions indépendamment les uns des autres avant de fusionner l'ensemble des décisions. Étant donné que plusieurs décisions prises sur des régions peuvent être négatives (pas correctes), il est avantageux de pouvoir rejeter l'objet en question avec le moins possible de calculs. La figure... illustre l'architecture de la cascade.

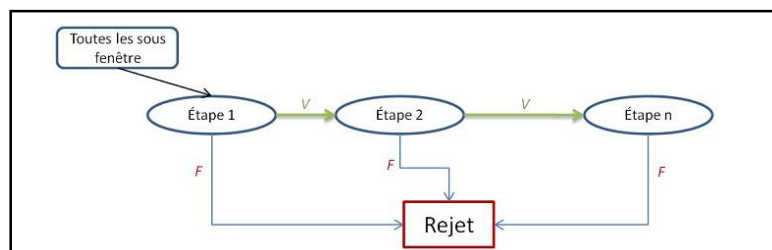


Figure1.3 : l'architecture d'un classifieur en cascade [2]

Comme montre la figure, la cascade est constituée d'une succession d'étages. En pratique, chacun est formé d'un classifieur fort appris par l'algorithme AdaBoost [30]. Le principe général de cet algorithme est de combiner un ensemble de classifieurs "faibles" (plus simples et plus rapides situés au début) en un classifieur fort (derniers classifieurs performants qui traitent les cas difficiles). La reconnaissance d'un objet par cascade de classifieurs est souvent à la fois performante et rapide. En revanche, l'inconvénient majeur de son implémentation est qu'elle requiert un temps d'apprentissage très élevé.

3.4 Défis de l'apprentissage du classifieur :

Les propriétés qui sont propres à la tâche d'apprentissage dans la détection d'objets, à savoir: l'échelle, la vitesse et l'asymétrie, font l'étape d'apprentissage du classifieur très difficile.[11]

- a) **défi de l'échelle** : Le nombre d'exemples d'entraînement nécessaires pour une tâche d'apprentissage augmente avec la complexité du problème. La détection de la classe d'objets est une tâche avec une grande complexité où il est nécessaire de détecter l'objet sous de nombreuses variantes telles que la pose, l'éclairage, déformation, etc. En conséquence, la base doit être assez grande, allant des dizaines de milliers à plusieurs milliards.
- b) **défi de la vitesse** : le temps d'apprentissage et d'entraînement d'un classifieur peut être influencé par la taille de la base d'apprentissage, d'où le temps nécessaire augmente autant le nombre d'exemples dans la base. Cependant, ceci n'influe pas sur le temps mais aussi sur la vitesse de détection lors la phase de test ou la détection en ligne. En effet, la vitesse de classifieur est plus importante pour la détection d'objet.
- c) **Le Défi d'asymétrie**: la détection d'objet comprend une tâche d'apprentissage fortement asymétrique. La propriété d'asymétrie contribue grandement à la difficulté de l'apprentissage du classificateur. Trois asymétries sont résumées: priors de classe, objectif asymétrie et complexité inégale au sein des classes positives et négatives.

4. Approches de détection d'objets : Etat de l'art

Après l'étude du problème de classification, nous allons entamer dans l'état de l'art sur les différents travaux connexes à la comparaison entre les classifieurs dans le domaine de détection d'objets, et notamment ceux qui appliqués dans un environnement routier.

Dans la littérature, nous avons trouvé peu de travaux autour de la détection d'objets dans un environnement routier. Pour mieux extraire des informations utiles à la comparaison entre ces différents travaux, nous les regroupons dans le tableau comparatif ci-dessous. Nous avons utilisé les même critères de comparaison que mentionnés dans la référence [31].

Nom, Titre et référence de méthode évaluée	Critères d'évaluation				
	Nombre de classes et exemples d'entraînement	Taux de Reconnaissance	Coût Informatique	Descripteur utilisé	Méthode de classification
Kato and Ninomiya: "Preceding vehicle recognition based on learning from sample Images" [33]	2 classes: 5000 véhicules et 5000 non-véhicules exemples à entraîner	TBR : 96%-98%	Reconnaissance: 0,5 - 1,7 ms @ PIII 800 MHz	sans autre précision	MC-MQDF - Classifieur Linéaire
Sun e.a. : « Monocular pre-crash vehicle detection: Features and classifiers » [30]	2 classes: 1051 véhicule, 1051 non véhicules	Taux d'erreur : 3.8%-9.1%	100 ms @ PIII 1133MHz	PCA, ondelette, tronquée / quantifiée ondelettes, caractéristiques de Gabor et une combinaison	réseaux de neurones & machines à Vecteur de support
Opelt e.a. : " Generic object recognition with boosting " [31]	3 classes: 450 personnes, 350 vélos 250 non vélo / personne	TBR : 6,5 à 83,5%	non précisé	moments d'intensité, SIFT	AdaBoost
Handmann, Kalinke, e.a. : " An image processing system for driver assistance " [34]	2 classes: véhicule / non véhicule; traitement entièrement automatique	non spécifié	reconnaissance: ≈ 50 ms @ DEC Alpha 500 MHz	caractéristiques de forme, symétrie locale, texture, ombres et couleurs	LOC-classifieur, Hausdorff classifieur
Laetitia Leyrit : « Reconnaissance d'objets en vision artificielle: application a la reconnaissance de piétons» [11]	2 classes: piétons /non piétons	non spécifié	non spécifié	ondelettes de Haar et histogrammes de gradients	Classifieur Adabost

Bassem Besbes : « Intégration de méthodes de représentation et de classification pour la détection et la reconnaissance d'obstacles dans des scènes routières » [10]	3 classes : Piétons et Véhicule et Fond d'images	Locales : 63.94% Globales ; 87.41%	non spécifié	SIFT et SURF	SVM
R. Muralidharan et al[]	100 database	82.50% 75%	non spécifié	SIFT	KNN et SVM
Constantine Papageorgiou and Tomaso Poggio, "A Trainable System for Object Detection" [36]	2 classes: piétons et non piétons	non spécifié	non spécifié	Haar	SVM
Vahid Balali, Elizabeth Depwe et Mani Golparvar-Fard, "Multi-class Traffic Sign Detection and Classification Using Google Street View Images" [35]	4 classes : 1523 classe1/ 5924 classe 2/ 1640 classe 3/ 809 classe4	TBR: 90.75% for HOG and 95.5% for HOG+C	Intel(R) Core(TM) i7-3820 CPU @ 3.60 GHz with 64.0 323 GB RAM and NVIDIA GeForce GTX 400 graphics card.	HOG et HOG+c	SVM (un contre tous)
Abdelhamid Mammeri, Depu Zhou, et Azzedine Boukerche « Animal-Vehicle Collision Mitigation System for Automated Vehicles" [37]	2 classes : animal et non-animal 3462 animaux et 7000 non-animaux	Miss rate < 0.25	Reconnaissance: 64.32 ms Intel core i5-2450M 2.50 GHz dual-core with 4 GB of RAM.	HOG , LBP et Haar	AdaBoost et SVM (classifieur en cascade)

Tablea2.2: tableau récapitulatif de différents travaux de recherches

4.1 Bilan

D'après le tableau ci-avant, nous synthétisons que la plupart des travaux évaluent la performance des systèmes de détection en utilisant des classifieurs binaires (seulement 02 classes concernées par la catégorisation (objet d'intérêt/autres), tandis que seulement peu de travaux qui ont tenté d'utiliser des classifieurs multiclassés (le cas des travaux [35], [10] et [31]). Du même, ces derniers travaux ciblent plusieurs classes de même catégorie (cas de travail [35]) où le problème de variation intra-classe peut influencer la performance du classifieur.

D'une autre part, les travaux présentés utilisent aussi trois variantes de descripteurs de caractéristiques : un seul ou une combinaison de descripteurs locaux, un ou une combinaison de descripteurs globaux ou bien une combinaison des descripteurs locaux et globaux. Ces variantes peuvent être testées avec un seul ou plusieurs classifieurs.

Finalement nous observons que la performance de systèmes de détection ou de classifieurs peut être jugée en deux points de vue :

* l'exactitude de classification : ce point est défini par le taux de bonne reconnaissance maximum ou le taux de fausse alarme (miss rate) minimum. D'après le tableau nous remarquons que cette mesure est variée selon les points suivants :

- Nombre de classes ciblées.
- Taille de dataset (nombre d'exemples positifs et négatifs d'entraînement dans la base d'apprentissage).
- Les descripteurs de caractéristiques ainsi que les méthodes de classification utilisées.

* le temps de reconnaissance : en plus des points susmentionnés, ce point est varié aussi selon le matériel (hardware) dans lequel l'entraînement et le teste ont été effectués.

5. Conclusion

Dans ce chapitre nous avons essayé d'exposer les différentes méthodes utilisées dans la description et la classification des objets et notamment celles de la classification supervisée. Ces méthodes ont connu beaucoup de progrès ces dernières années. Et pour l'état de l'art nous avons essayé de présenter quelques travaux antérieurs qui sont utilisés ces méthodes de classification.

Dans le chapitre suivant, nous allons présenter les métriques d'évaluation de performance de ces méthodes de classification, puis nous allons présenter une étude comparative entre ces méthodes selon ces métriques.

Chapiter 03

Analyse comparative des performances

1. Introduction

L'évaluation des performances d'un système de classification est une étape très importante pour quantifier son fonctionnement d'une part, et optimiser les valeurs des paramètres du classifieur d'une autre part. La performance se mesure avec des critères qualitatifs ou quantitatifs de résultat. Pour évaluer la performance d'un classifieur, il est nécessaire d'effectuer des mesures selon plusieurs critères. Généralement, les critères retenus sont l'exactitude de classification, ou le taux de bonne classification ou reconnaissance, c'est-à-dire le nombre d'éléments d'une base de test correctement classés, ainsi que la rapidité de classification. Néanmoins, un classifieur moins performant a un coût plus important pour la fiabilité du système dans certain environnement hostile. Par exemple, dans un environnement routier une mauvaise détection des obstacles peut, en effet, avoir différents coûts pour la sécurité routière.

Dans ce chapitre, nous dressons un panorama des métriques de performance des systèmes de classification à deux classes (classification binaire) puis de manière plus générale des systèmes multi-classes. Ensuite nous nous essayons de faire une étude expérimentale sur les performances de deux classifieurs supervisés connus avec des noyaux différents. Enfin nous discutons les résultats obtenus afin d'extraire les points de différence entre les classifieurs évalués.

2. Métriques de performance d'un classifieur

Dans cette section, nous allons évoquer, les métriques qui permettent d'évaluer et d'analyser les performances d'un système de classification (binaire et multiclassés).

2.1 Classifieur binaire [2]

Pour évaluer des classifieurs à deux classes (binaires), les résultats d'expérimentation sont remplis dans une matrice appelée « matrice de confusion ».

En effet, à partir de l'ensemble de test qui contient des exemples positifs correspondent à la classe d'intérêt (par exemple animal) et les exemples négatifs (autres objets que ne représentent pas des animaux « non-animal ») connus à priori, le classifieur va traiter ces exemples afin de reconnaître et identifier (prédire) leur catégorie (classe d'appartenance).

Par conséquent, à partir de décision du classifieur, les valeurs possibles qu'une matrice de confusion peut les contenir sont (voir le tableau....) :

- Vrais positifs (VP): c'est le nombre d'exemples positifs en réalité que le classifieur prédit son appartenance à la classe d'intérêt (prédiction positive).
- Faux positifs (FP): le nombre d'exemples négatifs en réalité que le classifieur prédit son appartenance à la classe d'intérêt (prédiction positive).
- Faux négatifs (FN): le nombre d'exemples positifs en réalité que le classifieur prédit son appartenance de façon incorrecte (identifier comme un autre objet non-animal « prédiction négative »).
- Vrais négatifs (VN): le nombre d'exemples négatifs en réalité que le classifieur prédit son appartenance à la classe autre que l'objet d'intérêt (prédiction négative).

		Réalité	
		Positifs	Négatifs
Prédictions	Positifs	Vrais positifs (VP)	Faux positifs (FP)
	Négatifs	Faux Négatifs (FN)	Vrais Négatifs (VN)

Tableau 3.1: Matrice de confusion pour la classification binaire

Pratiquement, les métriques de performance, qui repose sur la matrice de confusion, sont :

- **précision** : la précision est la quantité qui représente la capacité du classifieur à associer les exemples à la bonne classe:

$$\textit{Precision} = \frac{VP}{VP + FP}$$

- **rappel** : le rappel souligne la capacité du classifieur à retrouver les exemples positifs

$$\textit{Rappel} = \frac{VP}{VP + FN}$$

- **Accuracy (exactitude)**

$$\textit{Accuracy} = \frac{VP + VN}{VP + FN + FP + VN}$$

- **Sensibilité (Sensitivity)**

$$\textit{Sensitivity} = \frac{VP}{VP + FN}$$

- **spécificité (Specificity)**

$$\textit{Specificity} = \frac{VN}{FP + VN}$$

En outre, des courbes sont couramment utilisées pour évaluer les performances d'un apprentissage. Les plus courantes sont :

- **les courbes de précision/rappel :**

Les courbes de précision/rappel représentent la précision en fonction du rappel pour différents seuils de classification. Le cas idéal à atteindre est d'avoir une précision et un rappel de 100%, ce qui signifie que le classifieur est capable de retrouver tous les exemples positifs sans aucune fausse détection.

- **les courbes ROC :**

Les courbes ROC représentent le taux de bonnes détections en fonction du taux de fausses alarmes, soit les vrais positifs en fonction des faux positifs. Pour tracer cette courbe, il faut faire varier le seuil de décision et, pour chaque valeur, calculer le taux de bonnes détections et de fausses alarmes.

En outre, il faut également chercher à limiter le nombre de fausses alarmes, c'est-à-dire les faux positifs.

2.2 classification multiclassées [44]

		Décision			
		ω_1	ω_2	...	ω_C
Étiquette	ω_1	$Cost_{1,1}$	$Cost_{1,2}$...	$Cost_{1,C}$
	ω_2	$Cost_{2,1}$	$Cost_{2,2}$...	$Cost_{2,C}$
	\vdots	\vdots	\vdots	\ddots	\vdots
	ω_C	$Cost_{C,1}$	$Cost_{C,2}$...	$Cost_{C,C}$

Tableau 3.2 : Matrice de confusion multi-classes

- **Taux global de bonne reconnaissance (TGBR)**

(la somme de prototypes bien classées (diagonale)/somme totale) * 100

D'une autre façon, le TGBR et le taux d'erreur sont calculés comme suit :

$$- \text{TGBR} = \sum_{i=1}^L \frac{tp_i + tn_i}{tp_i + fn_i + fp_i + tn_i}$$

$$- \text{Taux d'erreur} = \sum_{i=1}^l \frac{fp_i + fn_i}{tp_i + fn_i + fp_i + tn_i}$$

Pour de nombreuses classes C_i :

$$\begin{cases} tp_i : \text{vraies positifs} \\ fp_i : \text{faux positifs} \\ fn_i : \text{faux négatifs} \\ tn_i : \text{Vraies négatifs} \end{cases}$$

3. Etude expérimentale des performances de classifieurs

Dans ce qui suit, nous allons comparer les performances des 2 méthodes de classification à noyau tout en évaluant avec certains paramètres (valeurs de noyau):

- SVM avec noyau de type " linear", noyau " polynomial " avec paramètre $p = \{2, 4\}$ et ainsi SVM avec noyau de type " RBF (radial basis function) " avec paramètre $\sigma = \{2, 4\}$.
- KPP avec $K=1$ et $k=3$.

Pour cela on détermine tous d'abord la base d'image qui nous avons utilisé dans ce cas puis on calcule la matrice de confusion pour détermine le taux de reconnaissance de chaque classe et le taux de reconnaissance global de chaque classifieur.

Nous notons que nous avons utilisé le MATLAB pour extraire ces résultats, les autres résultats de la matrice sont calculés manuellement.

3.1 Description de l'environnement MATLAB

MATLAB (« matrix laboratory ») [40] est un langage de programmation de quatrième génération émulé par un environnement de développement du même nom ; il est utilisé à des fins de calcul numérique. Développé par la société The MathWorks, MATLAB permet de manipuler des matrices, d'afficher des courbes et des données, de mettre en œuvre des algorithmes, de créer des interfaces utilisateurs, et peut s'interfacer avec d'autres langages comme le C, C++, Java, et Fortran. Les utilisateurs de MATLAB sont de milieux très différents comme l'ingénierie, les sciences et l'économie dans un contexte aussi bien industriel que pour la recherche. Dans ce travail, nous avons utilisé la version 2015 et nous avons essayés d'exécuter les différentes implémentations de SVM et KNN pour trouver les résultats qui nous sommes besoin afin de remplir la matrice de confusion.

3.2 Base d'images (dataset)

Pour réaliser l'opération de classification nous avons besoin de présenter le contenu de la base d'images utilisée pour l'apprentissage hors ligne et les tests.

Tous d'abord, notre base est formée par des images téléchargées depuis « GOOGLE IMAGE ». Elle est constituée de 120 images, classé en trois classes où chacune est décomposée en deux parties, l'une pour l'apprentissage et l'autre pour faire le test.

Dans notre étude les trois classes d'intérêt ciblées sont :

- **Classe Piéton** : elle représente les personnes qui peuvent traverser une route. Dans cette class nous avons fournie 30 images: 20 pour l'apprentissage et 10 pour le test.
- **Classe Véhicule** : peut contenir trois types (voitures, motos et bus). Dans cette class nous avons fournie 50 images: 20 pour l'apprentissage et 30 pour le test.
- **Class Animal** : sont tous les animaux qui nous peuvent trouver dans une route, comme : les chameaux, les chiens ou les chats ...etc. Dans cette class nous avons fournie 40 images: 20 pour l'apprentissage et 20 pour le test.

3.3 Evaluation de performance

a) La méthode SVM multiclassés

Nous évaluons un classifieur SVM multi-classes en utilisant l'approche un contre tous pour différents noyaux et différents paramètres des noyaux. Les matrices de confusion obtenues sont reportées dans les tableaux suivants :

- **Linear :**

	Véhicule	animal	piétons
Véhicule	26.67	23.33	50
Animal	5	70	25
Piétons	0	10	90

51.667%

Tableau 3.3: Matrice de confusion de la classification multiclassés pour un noyau linear

- Le taux de reconnaissance global est : 51.667%
- Le taux d'erreur global est: 48.333%

- **RBF (Radial Basis Function) ($\sigma = 2$):**

$\sigma = 2$	Véhicule	Animal	piétons
Véhicule	63.33	23.33	13.33
Animal	35	50	15
Piétons	40	10	50

56.667%

Tableau 3.4: Matrice de confusion de la classification multiclassés pour le noyau RBF ($\sigma = 2$)

- Le taux de reconnaissance global est : 56.667%
- Le taux d'erreur global est: 43.333%

- **RBF (Radial Basis Function) ($\sigma = 4$):**

$\sigma = 4$	Véhicule	Animal	Piétons
Véhicule	60	33.33	6.66
Animal	30	60	10
Piétons	40	10	50

58.333%

Tableau 3.5: Matrice de confusion de la classification multiclassées, noyau RBF ($\sigma = 4$)

- Le taux de reconnaissance global est : 58.333%
- Le taux d'erreur global est: 41.667%

- **Polynomial ($p=4$):**

P=4	Véhicule	animal	Piétons
Véhicule	10	40	50
Animal	20	55	25
Piétons	10	20	70

35%

Tableau 3.6: Matrice de confusion de la classification multiclassées pour le noyau Polynomial ($p=4$)

- Le taux de reconnaissance global est : 35%
- Le taux d'erreur global est: 65%

- **Polynomial ($p=2$) :**

P=2	Véhicule	Animal	piétons
Véhicule	20	26.667	53.33
Animal	20	60	20
Piétons	10	20	70

41.6667%

Tableau 3.7: Matrice de confusion de la classification multiclassées pour le noyau Polynomial ($p=2$)

- Le taux de reconnaissance global est : 41.6667%
- Le taux d'erreur global est: 58.3333%

b) La méthode KNN multiclassées

- Pour K=1

K=1	véhicule	Animal	piétons
Véhicule	40	26.66	33.33
Animal	15	50	35
Piétons	30	10	60

46.6667%

Tableau 3.8: Matrice de confusion de la classification multiclassées pour knn(K=1)

- Le taux de reconnaissance global est : 46.6667%
- Le taux d'erreur global est: 53.3333%

- Pour K=3

K=3	véhicule	Animal	Piétons
Véhicule	43.33	26.667	30
Animal	20	60	20
Piétons	20	10	70

53.333%

Tableau 3.9: Matrice de confusion de la classification multiclassées pour KNN(K=3)

- Le taux de reconnaissance global est : 53.333%
- Le taux d'erreur global est: 46.667%

3.4 Comparaison des résultats et discussions

- 1 ère cas (entre les noyaux de SVM):

Noyau	RBV		Polynomiale		Linéaire
	$\sigma = 4$	$\sigma = 2$	P=4	P=2	
véhicule	60	63.3333	10	20	26.6667
Animal	60	50	55	60	70
Piétons	50	50	70	70	90
TGBR	58.3333	56.6667	35	41.6667	51.6667

Tableau 3.10: Résumé des résultats pour la classification multiclassées avec la méthode SVM

Nous présentons dans cette figure suivante le taux de bonne de reconnaissance pour chaque classe avec la méthode SVM:

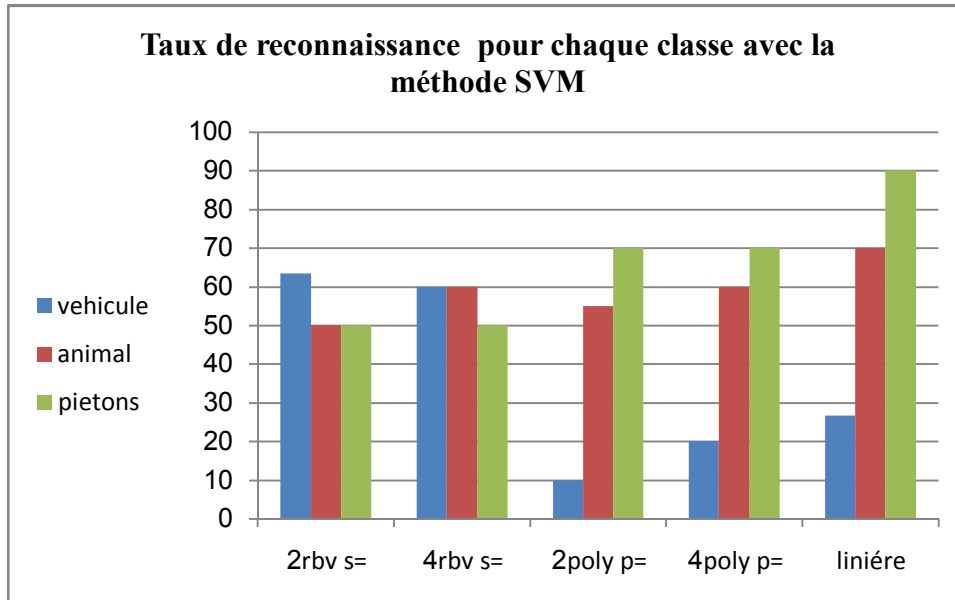


Figure 3.1: Taux de reconnaissance pour chaque classe avec la méthode SVM

D'après les résultats présentés dans le tableau ci-avant, nous constatons que le choix du noyau linéaire dans notre système fournit les meilleurs taux de bonne de reconnaissance comparativement aux autres noyaux pour les deux classes **animal** et **piéton** mais pour la classe **Véhicule** le meilleur taux de bonne reconnaissance est pour le choix de noyau RBF avec $\sigma = 2$ où les deux autre classes sont mal reconnues.

- **2 ème cas (entre les noyaux de KNN) :**

Noyau	K=1	K=3
Véhicule	40	43.33
Animal	50	60
Piétons	60	70
TGBR	46.667	53.333

Tableau 3.11: Résumé des résultats pour la classification multiclassés avec la méthode KNN

La figure suivante présente le taux de bonne de reconnaissance pour chaque classe avec la méthode KNN:

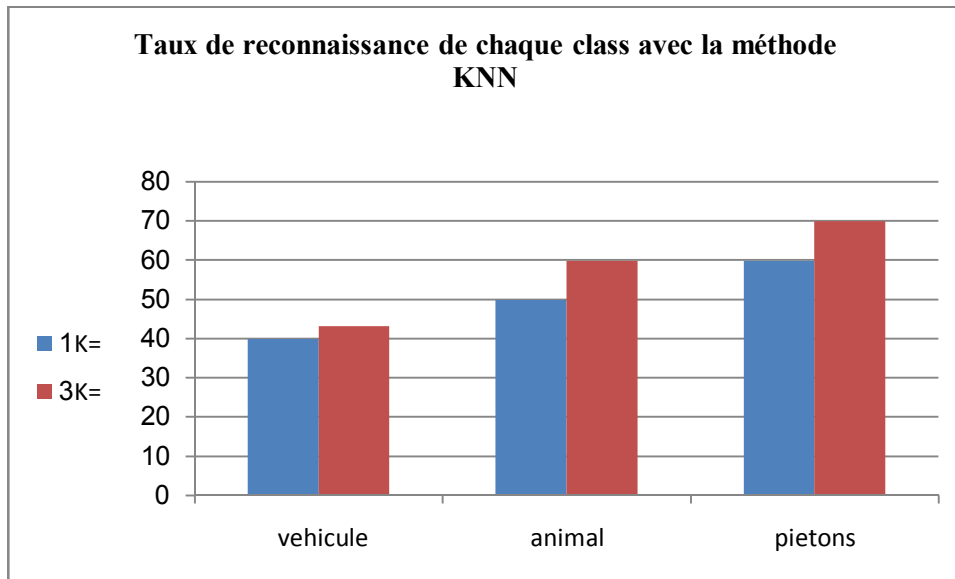


Figure 3.2: Taux de bonne reconnaissance pour la méthode KNN

D'après cette figure, nous observons que le choix de la valeur $K=3$ pour la méthode KNN fournit les meilleurs taux de bonne de reconnaissance par rapport aux autres valeurs et cela pour toutes les classes.

- **3ème cas (entre SVM et KNN):**

Noyon	Svm					knn	
	RBV		Polynomiale		Linéaire	K=1	K=3
	$\sigma = 4$	$\sigma = 2$	P=4	P=2	P=1		
véhicule	60	63.3333	10	20	26.6667	40	43.33
Animal	60	50	55	60	70	50	60
Piétons	50	50	70	70	90	60	70
TGBR	58.3333	56.6667	35	41.6667	51.6667	46.667	53.333

Tableau 3.12: Résumé des résultats pour la classification multiclassés avec KNN et SVM

La figure ci-dessous présente le taux global de bonne reconnaissance pour chaque méthode.

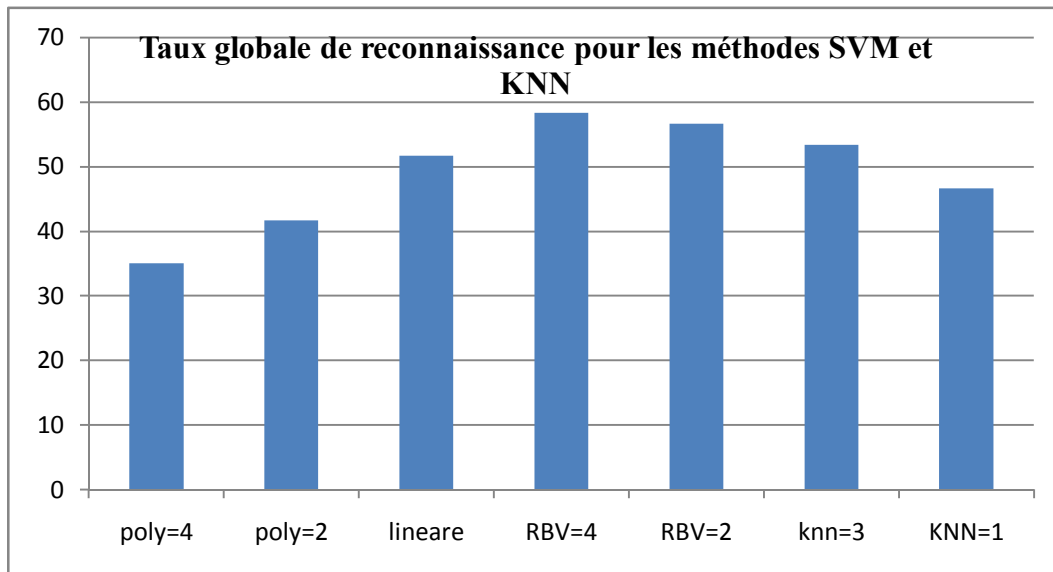


Figure 3.3 : Taux globale de bonne reconnaissance pour chaque méthode

D'après l'analyse des résultats obtenus (comme montré dans le tableau et la figure ci-avant), on remarque que, dans notre cas (environnement routier avec 3 classes ciblées), la méthode SVM avec le noyau RBF avec $\sigma = 4$ est la mieux pour la classification car elle fournit le meilleur taux de bonne reconnaissance global (pour toutes les classes). Cependant, cette méthode présente aussi un fiable taux de reconnaissance quant au noyau polynomiale (pour les deux valeurs du paramètre « p ») par rapport aux autres choix (noyaux et paramètres).

- **discussion générale:**

Généralement, on peut constater que les taux de bonne reconnaissance sont sensiblement similaires pour toutes les classes, où la méthode SVM avec le noyau RBF avec $\sigma = 4$ donne le plus grand taux de bonne reconnaissance global ainsi qu'elle donne le fiable taux de reconnaissance quant au noyau polynomiale, et variés selon la classe ciblée. Cette variation est due au choix de la méthode ainsi que ses paramètres (noyau et les paramètres associés). Ceci montre qu'il n'existe pas un noyau plus favorable (optimal).

En fait ; les résultats obtenus dépendent des classes et des différents facteurs comme la taille de base d'apprentissage utilisées, les descripteurs de caractéristiques, le choix de la méthode... etc. donc, nous concluons qu'il n'existe pas une méthode optimale pour tous les cas.

4. Conclusion

Dans ce chapitre, nous avons présenté de manière simple une comparaison expérimentale entre les deux méthodes de classification supervisée KNN et SVM (classification multi-classes) avec l'utilisation des différents noyaux. A partir de résultats obtenus nous avons conclu que la performance de chaque classifieur est dépend généralement aux plusieurs facteurs comme : la taille de base d'apprentissage ou celle de base de test, les type et les valeurs des noyaux utilisés, les types et le choix d'images qui sont utilisées pour l'apprentissage et le test ,et d'autre facteurs influent sur la performance de ces deux méthodes de classification .

Généralement il n'existe pas des règles fixées et précises pour choisir les valeurs optimales afin d'assurer une méthode de classification optimale.

Conclusion Générale

Dans le cadre ce travail, l'objectif est d'étudier le problème de classification multi-classes en apprentissage supervisé hors ligne. Précisément, de faire évaluer expérimentalement deux méthodes de classification utilisées dans les systèmes intelligents d'aide à la conduite (ADAS) basés sur détection d'objets dans le contexte des environnements routiers.

Au cours de ce mémoire, nous avons fait un panorama sur le problème de classification qui est une phase essentiel dans le processus de détection en ligne. Néanmoins, avant d'arriver à développer un système de reconnaissance classification, nous avons vu qu'il nécessite deux phases principales : la phase de représentation et la phase de classification. Cette dernière phase est constituée des autres phases : apprentissage hors ligne et décision.

Pour la réalisation de notre étude comparative, nous avons analysé les différent travaux de recherche autour les approches de détection d'objets est et notamment dans notre contexte. Puis, nous avons entamé notre étude avec les choix suivants: deux méthodes pour la classification multi-classes qui sont la méthode de séparateur à vaste marge SVM et la méthode K plus proche voisin KPP, ensuite, évaluons ces méthodes avec certains paramètres (différents noyaux et 3 classes). Pour cette étude nous avons utilisé le taux de bonne reconnaissance global de chaque classifieur et le taux de bonne reconnaissance de chaque classe comme métriques d'évaluation des performances de chaque méthode de classification. Après cette étude nous avons obtenu comme résultat que le classifieur SVM avec le noyau $\sigma=4$ est le plus performante dans notre cas mais si nous avons changé les situations comme la taille de base d'apprentissage ou la taille de base de test ou si nous avons changé le nombre de classe nous peuvent être obtenue autre résultat.

A la fin de ce mémoire, nous avons arrivé pratiquement à conclure qu'il n'existe pas une méthode fiable pour tous les cas. C'est-à-dire selon les conditions fournies, les résultats de classification et les performances des classifieurs sont changées.

Finalement, nous proposons d'améliorer le présent travail, comme perspective, par:

- Le Développement d'un système de détection et reconnaissance en ligne basé sur méthode de classification multiclassées.
- L'étude des méthodes de classification non supervisée et de faire une autre étude comparative entre eux avec les méthodes de classification supervisée.

Bibliographie

- [01] mémoire de *GÉNÉRALITÉS SUR LA CLASSIFICATION*
- [02] Laetitia Leyrit. Reconnaissance d'objets en vision artificielle : application à la reconnaissance de piétons. Sciences de l'ingénieur [physics]. Université Blaise Pascal - Clermont-Ferrand II, 2010.
- [03] vision humain , wikipedia, [En ligne]: http://fr.wikipedia.org/wiki/Vision_humaine
- [04] vision par ordinateur, https://fr.wikipedia.org/wiki/Vision_par_ordinateur
- [05] Christopher J.C.H. Watkins and Peter Dayan. Technical note : Q-learning. *Machine Learning*, 8(3-4) :279–292, May 1992.
- [06] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern classification* . John Wiley & Sons, 2001
- [07] Anil K. Jain and Richard C. Dubes. *Algorithms for clustering data*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1988.
- [08] Pr. Frédéric JURIE et Jean-Luc MAILLART et Pr. Roger MOHR Reconnaissance de catégories d'objets et d'instances d'objets à l'aide de représentations locales l'Ecole Doctorale Mathématiques, Sciences et Technologie de l'Information le 17 Mars 2008
- [09] détection d'objet , https://fr.wikipedia.org/wiki/D%C3%A9tection_d%27objet
- [10] Bassem Besbes. Intégration de méthodes de représentation et de classification pour la détection et la reconnaissance d'obstacles dans des scènes routières. *Computer Science*. INSA de Rouen, 2011. French. <NNT : 2011ISAM0007>. <tel-00633109>
- [11] Jianxin Wu and James M. Rehg, ”**Chapter 8: Object Detection**”, livre “*Ensemble Machine Learning: Methods and Applications*”, Springer Science+Business Media©, 2012.
- [12] Xin Zhang, Yee-Hong Yang, Zhiguang Han, Hui Wang, and Chao Gao,” **Object class detection: A survey**”, *ACM Comput. Surv.* 46, 1, Article 10 (July 2013), 53 pages.
- [13] Melle Maamra OumElhana et Melle Settou Trablesse Proposition d'un modèle de descripteur structurel pour la voix arabe, Application saisie des notes
- [14] Franky Y.SHIH , *IMAGE PROCESSING AND PATTERN RECOGNITION*, IEEE
- [15] Kristen Grauman and Bastian Leibe, “**Visual Object Recognition**”, *Synthesis Lectures on Artificial Intelligence and Machine Learning*, April 2011, Vol. 5, No. 2 , Pages 1-181

- [16] Richard Szeliski, *Computer Vision: Algorithms and Applications* September 3, 2010 draftc 2010 Springer
- [17] D. G. Lowe, “**Object recognition from local scale-invariant features**”, In *Proceedings of International Conference on Computer Vision*, pages 1150–1157, 1999.
- [18] H. Bay, T. Tuytelaars, and L. Van Gool, “**Surf: Speeded up robust features**”, In *European Conference on Computer Vision*, May 2006
- [19] Dalal, N., Triggs, B.: “**Histograms of oriented gradients for human detection**”. In: *Proc. of the IEEE Conference on CVPR*. vol. 2., San Diego, CA, USA. pp. 886–893 (2005)
- [20] Agarwal, S., Roth, D.: “**Learning a sparse representation for object detection**”. In: *Proc. European Conf. Comp. Vision*. Volume LNCS 2353., Copenhagen, Denmark (2002), 113–127
- [21] Fergus, R., Perona, P., Zisserman, A.: “**Object class recognition by unsupervised scale-invariant learning**”. In: *Proc. IEEE Conf. Comp. Vision Patt. Recog. Vol. II* (2003) 264–271
- [22] Fei-Fei, L., Fergus, R., Perona, P.: “**Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories**”. In: *Proc. IEEE Conf. Comp. Vision Patt. Recog Workshop on GenerativeModel Based Vision*. (2004)
- [23] M. Everingham, L. Van Gool, C.K.I. Williams, J. Winn and A. Zisserman, “**The PASCAL Visual Object Classes (VOC) Challenge**”, *Int'l J. Computer Vision*, vol. 88, no. 2, pp. 303-338, June 2010.
- [24] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., and Fei-Fei, L. “**ImageNet large scale visual recognition challenge**”. CoRR, abs/1409.0575, 2014.
- [25] J. Ponce, T. L. Berg, M. Everingham, D. A. Forsyth, M. Hebert, S. Lazebnik, M. Marszalek, C. Schmid, B. C. Russell, A. Torralba, C. K. I. Williams, J. Zhang, and A. Zisserman, “**Dataset issues in object recognition**». In J. Ponce, M. Hebert, C. Schmid, and A. Zisserman, editors, *Toward Category-Level Object Recognition*, volume 4170 of LNCS, pages 29–48. Springer Verlag, 2006.
- [26] L. LEYRIT, « **Reconnaissance d’objets en vision artificielle : Application à la reconnaissance de piétons** », Thèse de doctorat, École Doctorale Sciences Pour l’Ingénieur de Clermont-Ferrand, 2010.
- [27] Cortes, C., & Vapnik, V. , “**Support-vector networks**”, *Machine learning*,20(3), 1995, 273-297.
- [28] R. O. Duda and P. E. Hart, “**Pattern Classification and Scene Analysis**”. New York: John Wiley & Sons, 1973.
- [29] M. Jones, P. Viola, M. J. Jones, and D. Snow, “**Detecting pedestrians using patterns of motion and appearance**”, *IEEE International Conference on Computer Vision*, pages 734–741, 2003.
- [30] Y. Freund and R. Schapire, “**Experiments with a new boosting algorithm**”, *International Conference on Machine Learning*, pages 148– 156, 1996.

- [31] A. Laika and W. Stechele, "**A review of different object recognition methods for the application in driver assistance systems**," *Image Analysis for Multimedia Interactive Services, 2007. WIAMIS '07. Eighth International Workshop on*, Santorini, 2007, pp. 10-10.
- [32] MATHIEU-DUPAS **Algorithme des K plus proches voisins pondérés (WKNN)et Application en diagnostic** Eve MATHIEU-DUPAS Responsable Plate-forme bioinformatique & biostatistique UMR SysDiag, Modélisation et ingénierie des systèmes complexes biologiques pour le diagnostic
- [33] Takeo Kato, Yoshiki Ninomiya, and Ichiro Masaki, "**Preceding vehicle recognition based on learning from sample images**," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 3, no. 4, Dec 2002, pp. 252–260.
- [34] Uwe Handmann, Thomas Kalinke, Christos Tzomakas, Martin Werner, and Werner v. Seelen, "**An image processing system for driver assistance**," *Image and Vision Computing*, vol. 18, no. 5, pp. 367–376, Apr 2000.
- [35] Balali, V., Depwe, E., and Golparvar-Fard, M. "**Multiclass Traffic Sign Detection and Classification Using Google Street View Images**." Transportation Research Board 94th Annual Meeting, TRB, Washington, DC2015, USA.
- [36] C. Papageorgiou and T. Poggio, "A Trainable System for Object Detection," *International Journal of Computer Vision* 38(1), 15–33, 2000.
- [37] A. Mammeri; D. Zhou; A. Boukerche, "Animal-Vehicle Collision Mitigation System for Automated Vehicles," in *IEEE Transactions on Systems, Man, and Cybernetics: Systems* , vol.PP, no.99, pp.1-13,2016
- [38] B.Seddik , Descripteur SIFT, ISET Kairouan 2014-2015
- [39] discripteur surf , wikipedia, : [http:// fr.wikipedia.org/wiki/descripteur SURF](http://fr.wikipedia.org/wiki/descripteur_SURF)
- [40] MATLAB and Statistics Toolbox Release 2015b, The MathWorks, Inc., Natick, Massachusetts, United States.
- [41] Zehang Sun, George Bebis, and Ronald Miller, "**Monocular pre-crash vehicle detection: Features and classifiers**," *IEEE transactions on image processing*, vol. 15, no. 7, pp. 2019–2034, Jul 2006.
- [42] A. Opelt, A. Pinz, M. Fussenegger, and P. Auer, "**Generic object recognition with boosting**," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 28, no. 3, pp. 416–431, Mar 2006.
- [43] Histogrammes de gradients orientés, wikipedia, [En ligne]: [http:// fr.wikipedia.org/wiki/ Histogrammes de gradients orientés](http://fr.wikipedia.org/wiki/Histogrammes_de_gradients_orientés).

[44] YANNICK OUFELLA, ÉVOLUTION DU CONCEPT DE FRONT ROC
ET COMBINAISON DE CLASSIFIEUR, UNIVERSITÉ DE ROUENFR DES
SCIENCES ET TECHNIQUES MASTER GÉNIE INFORMATIQUE MASTER
GÉNIE ÉLECTRIQUE ET INFORMATIQUE INDUSTRIELLE 2 SEPTEMBRE
2008.

[Can07] S. Canu. Machines `a noyaux pour l'apprentissage statistique. *Techniques de l'ingenieur*, TE5255, 2007.

[HT98] T. Hastie and R. Tibshirani. Classification by pairwise coupling. *Annals of statistics*, 26(2) :451–471, 1998.

Annexe



Figure 4.1 : exemple d'images d'entraînement et de teste pour les classes : animal, piéton et véhicule dans notre dataset.

- **Étapes de l'algorithme de classification dans MATLAB**

Les différentes étapes de l'algorithme de classification sont les suivantes:

- **Étape n°1: La lecture d'une ou de plusieurs images**

- C'est l'étape initiale qui nous permet de créer l'ensemble d'apprentissage.

- **Étape n°2: La création de l'ensemble d'apprentissage**

- Il est constitué de l'ensemble de données libellées et non-libellées, représentant les pixels des images, cet ensemble est identifié à partir d'un superviseur.
- L'analyste, en utilisant un carré de pixels par classe (ex. un carré de pixels, pixels, etc.), va annoter les pixels de chaque classe, pour représenter l'ensemble, et le reste des pixels des images vont représenter l'ensemble.

- **Étape n°3: La création du vecteur de caractéristiques**

- Après avoir identifié l'ensemble d'apprentissage, l'étape suivante est de détecter puis extraire les caractéristiques tout en les conservant dans un vecteur appelé le vecteur de caractéristiques. (Il existe plusieurs façons pour constituer ce vecteur, en utilisant les descripteurs par exemple descripteur locale SURF).

- **Étape n°4 : La classification des données non-libellées**

- On va classifier les données de à l'aide d'un classifieur, en utilisant un des classifieurs de l'outil Matlab.
- Matlab dispose d'un ensemble de méthodes de classifications prédéfinis dans sa bibliothèque, et notamment:

- **Méthode de k-plus proche voisin :**

$[Class] = knnclassify(Sample U, Training L, Group)$

- **Méthode de SVM :**

$[Svmstruct] = svmtrain(Training L, Group)$

$[Group] = svmclassify(Svmstruct, Sample U)$

La sortie de l'algorithme : C'est l'ensemble d'apprentissage, qui va être utilisé pour classifier des données futures.

- Les interfaces de notre système

- 1) Résultats de paramétrage de noyau SVM

- a) noyau Gaussien (RBV) avec valeurs différentes de « sigma (δ) »

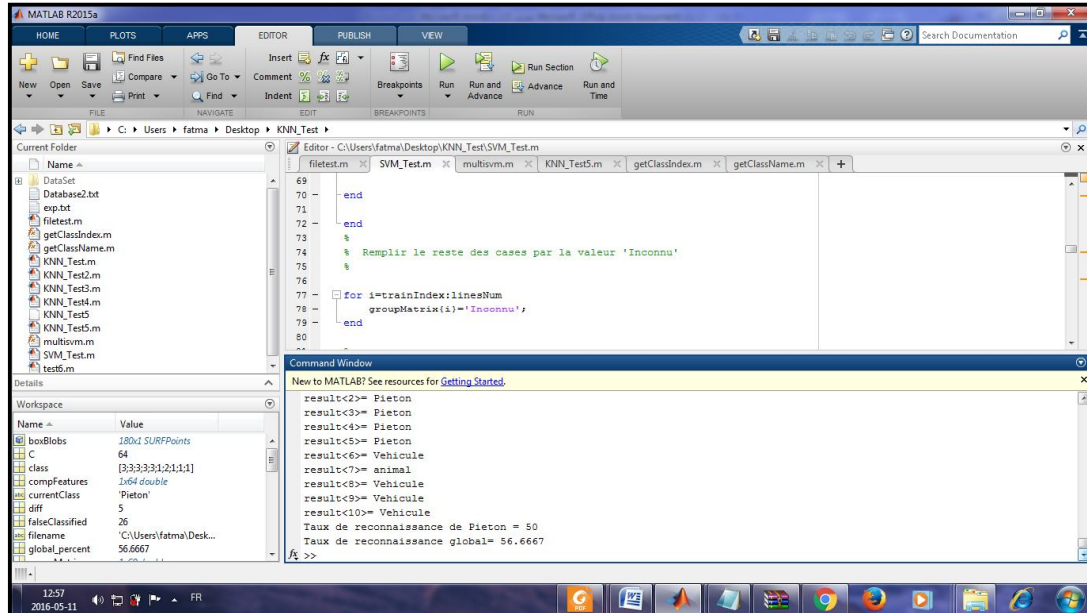


Figure4.4 : interface pour noyau RBV ($\delta = 2$)

- b) noyau Polynomiale (polynomial) avec valeurs différentes de « polyorder (p) »

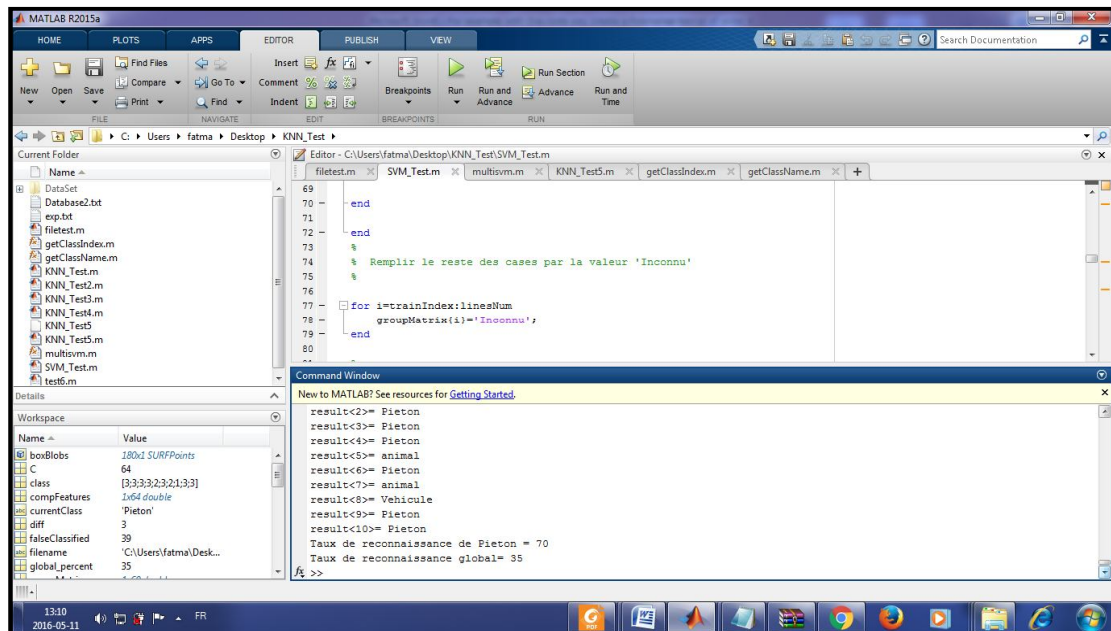


Figure4.5 : interface pour noyau polynomiale (p=4)

1) Résultats de paramétrage de noyau KNN

a) Pour K=1

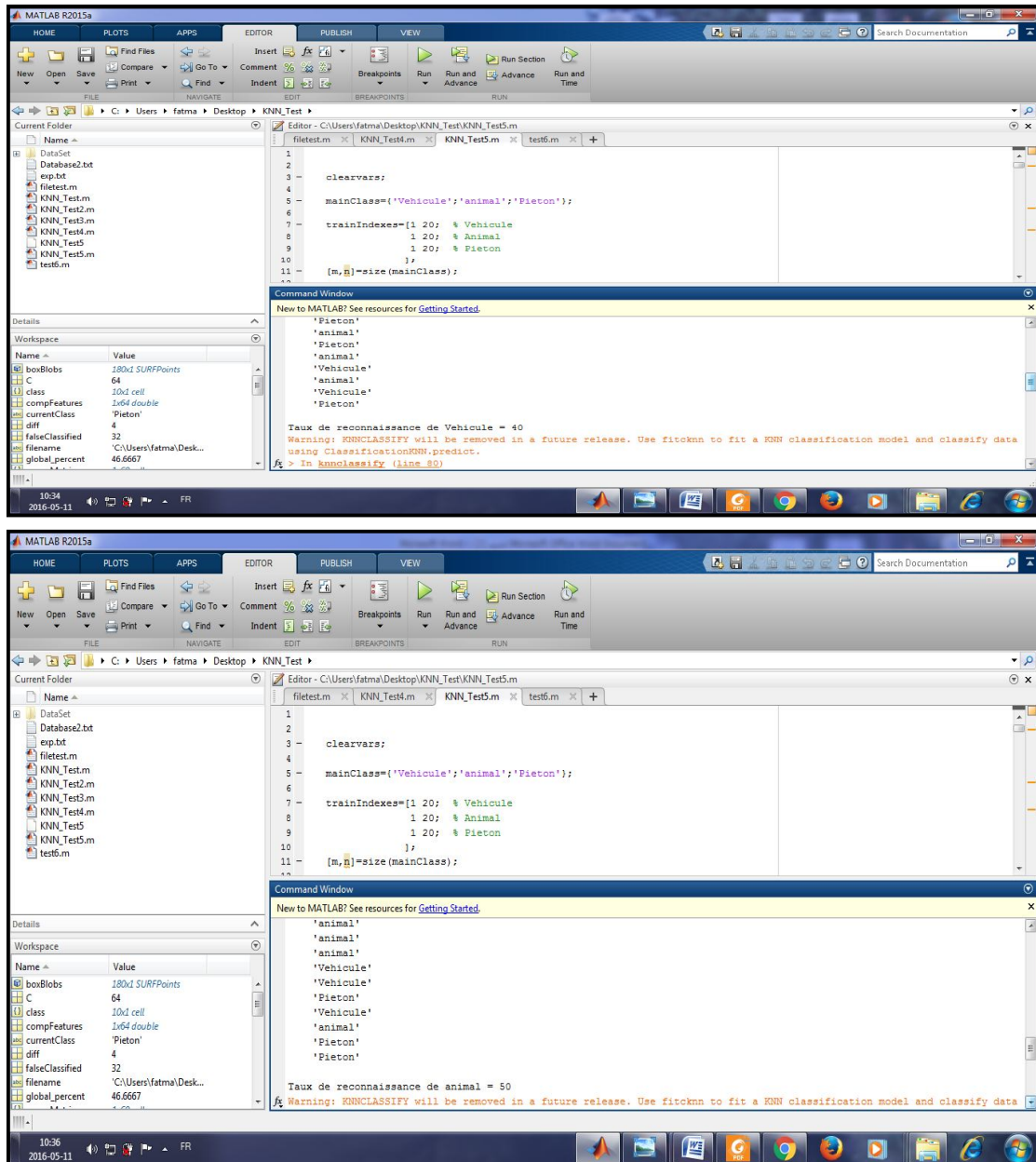
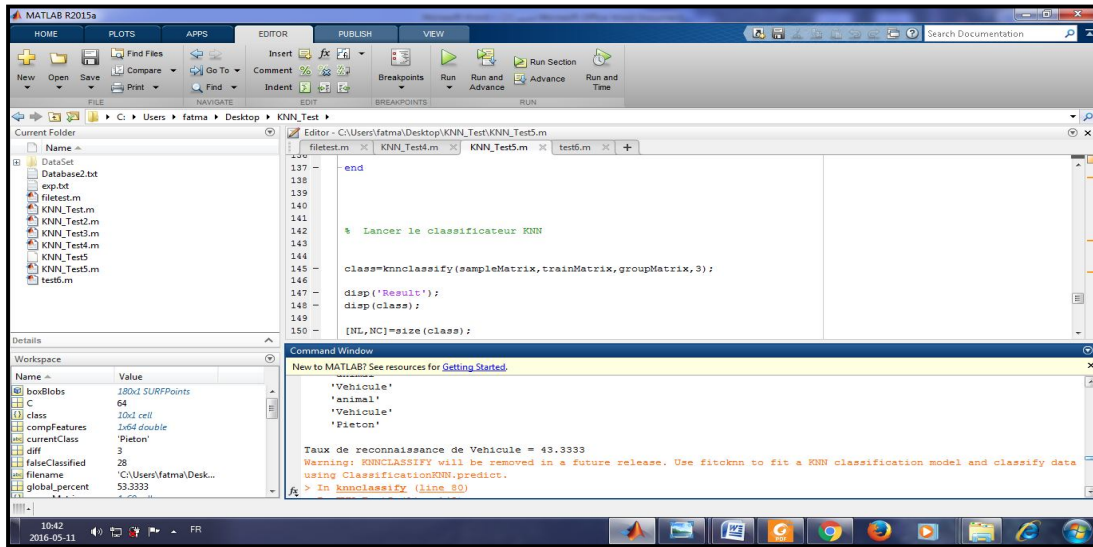


Figure4.6: interface pour KNN (K=1)

b) Pour $K=3$:

```
137 - end
138
139
140
141
142 % Lancer le classificateur KNN
143
144
145 class=knnclassify(sampleMatrix,trainMatrix,groupMatrix,3);
146
147 disp('Result');
148 disp(class);
149
150 [NL,NC]=size(class);
```

Command Window

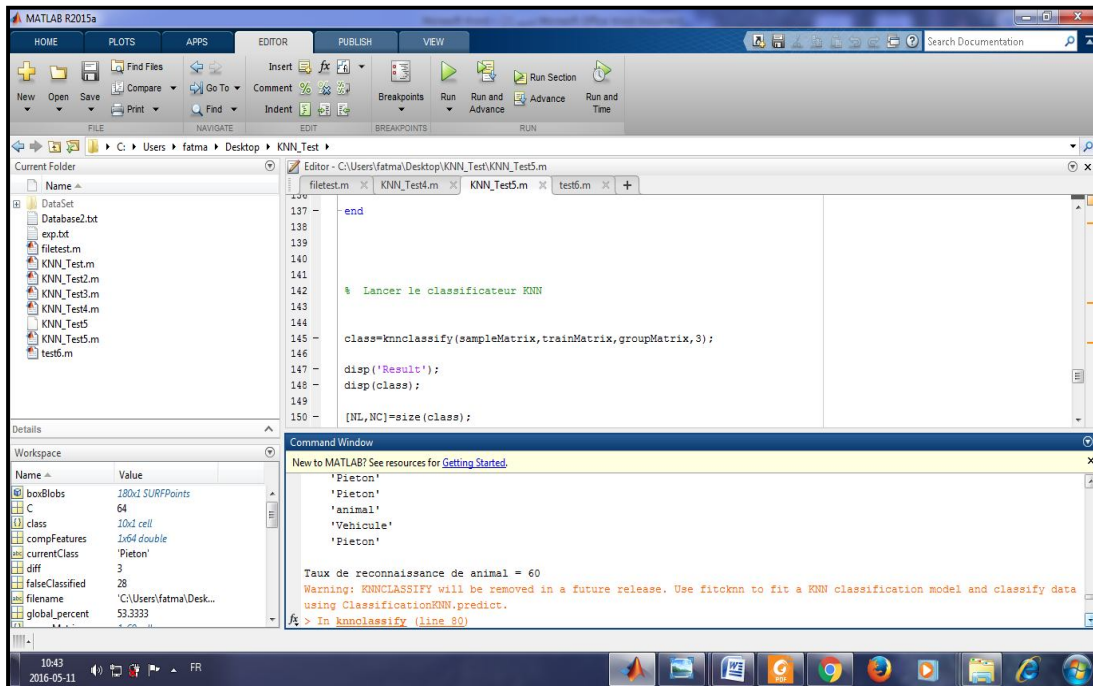
New to MATLAB? See resources for Getting Started.

```
'Vehicule'
'animal'
'Vehicule'
'Pieton'
```

Taux de reconnaissance de Vehicule = 43.3333

Warning: KNNCLASSIFY will be removed in a future release. Use fitknn to fit a KNN classification model and classify data using ClassificationKNN.predict.

In knnclassify (line 80)



```
137 - end
138
139
140
141
142 % Lancer le classificateur KNN
143
144
145 class=knnclassify(sampleMatrix,trainMatrix,groupMatrix,3);
146
147 disp('Result');
148 disp(class);
149
150 [NL,NC]=size(class);
```

Command Window

New to MATLAB? See resources for Getting Started.

```
'Pieton'
'Pieton'
'animal'
'Vehicule'
'Pieton'
```

Taux de reconnaissance de animal = 60

Warning: KNNCLASSIFY will be removed in a future release. Use fitknn to fit a KNN classification model and classify data using ClassificationKNN.predict.

In knnclassify (line 80)

Figure 4.7 : interface pour KNN ($K=3$)

- Codes sources

- Pour KNN classifieur:

```

clearvars;

mainClass={'Vehicule';'animal';'Pieton'};

trainIndexes=[1 20; % Vehicule
              1 20; % Animal
              1 20; % Pieton
              ];
[m,n]=size(mainClass);

linesNum=0;

for P=1:m
linesNum=linesNum+(trainIndexes(P,2)-
trainIndexes(P,1)+1);
end
groupMatrix=cell(1,linesNum);
trainMatrix=zeros(linesNum,64);

trainIndex=1;
for P=1:m
% Commencer l'apprentissage .
currentClass=char(mainClass(P));
for i=trainIndexes(P,1):trainIndexes(P,2)
filename=sprintf('C:\\Users\\fatma\\Desktop\\KNN_Test\\DataSet
\\%s_%d.jpg',char(mainClass(P)),i);
%disp(['Learning from image:',filename]);
Image = rgb2gray(imread(filename));
% Extraire les points ou les régions d'intérêt
ImagePoints=detectSURFFeatures(Image,'MetricThreshold', 2000);
%ImagePoints =detectMSERFeatures(Image);
% Extraire les caractéristiques
[ImageFeatures, boxBlobs]= extractFeatures(Image,ImagePoints);
% Compress features
[L,C]=size(ImageFeatures);
compFeatures=zeros(1,C);
som=0;
for k=1:C
for h=1:L
som=som+ImageFeatures(h,k);
end
compFeatures(k)=som;
som=0;
end
[NL,NC]=size(compFeatures);
groupMatrix{trainIndex}=currentClass;
for k=1:C

```

```
        trainMatrix(trainIndex, k)=compFeatures(k);
    end
    trainIndex=trainIndex+1;
end
end
% Remplir le reste des cases par la valeur 'Inconnu'
for i=trainIndex:linesNum
    groupMatrix{i}='Inconnu';
end
% Créer la matrice de test
mainClass={'Vehicule';'animal';'Pieton'};
testIndexes=[21 50; % Vehicule
             21 40; % animal
             21 30; % Pieton
             ];
[m,n]=size(mainClass);
global_percent=0;
rightClassified=0;
falseClassified=0;
for P=1:m
    targetClass=char(mainClass(P));
    sampleNum=testIndexes(P,2)-testIndexes(P,1)+1;
    sampleMatrix=zeros(sampleNum,64);
    testIndex=1;
    for i=testIndexes(P,1):testIndexes(P,2)
        filename=
        sprintf('C:\\Users\\fatma\\Desktop\\KNN_Test\\DataSet\\%
s_%d.jpg',char(mainClass(P)),i);
        Image = rgb2gray(imread(filename));
        % Extraire les points ou les régions d'intérêt
        ImagePoints =detectSURFFeatures(Image,'MetricThreshold', 2000;
        % Extraire les caractéristiques
        [ImageFeatures,boxBlobs]= extractFeatures(Image, ImagePoints);
        % Compress features
        [L,C]=size(ImageFeatures);
        compFeatures=zeros(1,C);
        som=0;
        for k=1:C
            for h=1:L
                som=som+ImageFeatures(h,k);
            end
            compFeatures(k)=som;
            som=0;
        end
        [NL,NC]=size(compFeatures);
        for k=1:C
            sampleMatrix(testIndex, k)=compFeatures(k);
        end
        testIndex=testIndex+1;
    end
end
```

```
% Lancer le classificateur KNN
class=knnclassify(sampleMatrix,trainMatrix,groupMatrix);
disp('Result');
disp(class);

[NL,NC]=size(class);
% Calculer le taux de reconnaissance
diff=0;
for i=1:NL
    if(strcmp(targetClass,class{i})==0)
        diff=diff+1;
    end
end
%diff
percent=((NL-diff)*100)/NL;
disp(['Taux de reconnaissance de ',targetClass,' =
',num2str(percent)]);
rightClassified=rightClassified+(NL-diff);
falseClassified=falseClassified+diff;
end
%clearvars;
somme=rightClassified+falseClassified;
global_percent=(rightClassified*100)/somme;

disp(['Taux de reconnaissance global=
',num2str(global_percent)]);
```