

Faculté des Sciences Exactes et de l'Informatique
Département de Mathématiques et d'Informatique
Filière : Informatique

MÉMOIRE DE FIN D'ETUDES
Pour l'Obtention du Diplôme de Master en Informatique
Option : **Systemes d'Information Géographique**

THÈME :
Reconnaissance des panneaux routiers
sur une image

Étudiants :

BENMAAZIZ Mohammed

GRINE Habib

Encadrant:

Mr. BESNASSI Miloud

Le but de ce travail est de concevoir, implémenter et tester une méthode de reconnaissance de panneaux routiers qui se base sur la vision par ordinateur (réseaux de neurones, support vecteurs machines ...).

L'approche adoptée dans ce travail est constituée de reconnaissance dont le rôle est de mettre en correspondance l'information visuelle détectée avec des modèles de panneaux routiers appris à priori. Nous allons d'abord essayer de déterminer quels sont les paramètres qui peuvent influencer sur les algorithmes de classification, tels que le nombre de couches cachées ou le pourcentage utilisé. Puis nous allons leur associer différents prétraitements, mesurer leurs effets sur le taux de reconnaissance ainsi que le temps mis pour afficher les résultats. Ces tests sont effectués sur un ensemble d'images réelles du trafic routier et montrent les performances du système actuellement développé.

Mots-clés : Traitement d'image, méthodes de classifications, réseaux de neurones, reconnaissance des formes, reconnaissance des panneaux routiers.

The goal of this work is to conceive, implement and test a method of detection and recognition of road panels which is based on the vision by computer (networks of neurons, support vectors machines, ...).

The approach adopted in this work consists recognition whose role is to put in correspondence the visual information detected with models of road panels learned a priori. We will first try to determine which parameters can influence the classification algorithms, such as the number of hidden layers or the percentage used. Then we go to their different pretreatments partner, measure their effects on the recognition rate as well as the time set to display the results. Tests carried out on a set of real images of the road traffic show the performances of the system currently developed.

Keywords: Method, image processing of classifications, network of neurons, Recognition of shapes, traffic sign recognition.

Je dédie ce modeste travail

À mes chers parents en témoignage de ma reconnaissance envers le soutien, les sacrifices et tous les Efforts qu'ils ont faits pour mon éducation ainsi que ma formation.

À mes chères sœurs

À toute ma famille

À ms chères amies « Madjid, Mohammed, Habib, Fatiha, Nora, Sara »

BENMAAZIZ Mohammed

Je dédie ce modeste travail

À mes chers parents en témoignage de ma reconnaissance envers le soutien, les sacrifices et tous les Efforts qu'ils ont faits pour mon éducation ainsi que ma formation.

À mes chers frères

À mes chères sœurs

À toute ma famille

À mes chères amies « Madjid, Mohammed, Aicha, Zohra »

GRINE HABIB

Remerciement

Tout d'abord, nous tenons à remercier ALLAH le Tout-Puissant et le miséricordieux, qui nous a donné la force et la patience d'accomplir ce modeste travail.

*Nous tenons à adresser également nos remerciements à notre encadreur Mr **BESNASSI Miloud** qui a bien voulu mettre leur incomparable savoir et expérience à notre disposition.*

Nos vifs remerciements vont également aux membres du jury pour l'intérêt qu'ils ont porté à notre recherche en acceptant d'examiner notre travail et de l'enrichir par à leur proposition.

Nous gardons une place toute particulière à nos parents, nos frères et nos sœurs qui sont toujours à nos côtés.

*Nous tenons aussi à remercier du fond du cœur **EL ARBI DAWADJI Mohammed** qui nous a aidés.*

Enfin, nous adressons nos plus sincères remerciements à tous nos proches et amis, qui ont toujours soutenu et encouragé au cours réalisation de ce mémoire.

Merci à tous et à toutes.

Résume	i
Abstract	ii
Dédicaces	iii
Remerciement.....	v
Sommaire	vi
Liste des tableaux	viii
Liste des figures	viii
Liste d'abréviations	ix
Introduction générale.....	1
1. Introduction.....	3
2. Définition de la reconnaissance des formes.....	3
3. Domaines d'application de la reconnaissance des formes	3
3.1. Reconnaissance des formes sur signaux temporels	3
3.2. Reconnaissance des formes dans les images numériques.....	3
4. Les Méthodes de la Reconnaissance des Formes	4
4.1. Approche statistiques :.....	4
4.2. Approche structurelle : (ou syntaxique)	4
5. Structure général d'un système de reconnaissance des formes.....	4
5.1. Acquisition et prétraitements.....	5
5.2. Extraction de caractéristiques.....	5
5.3. La classification.....	5
5.4. Apprentissage	5
5.5. La décision.....	7
6. Le but de la reconnaissance des formes	7
7. Tableau récapitulatif	7
8. Conclusion	7
1. Introduction :.....	8
2. Définition de la reconnaissance des panneaux (TSR) :	8
3. Les étapes de TSR.....	9
3.1. Prétraitement :.....	9
3.2. Extraction des caractéristiques :	9

3.2.1.	Le descripteur LBP (motifs binaires locaux) :	9
3.2.2.	Le descripteur ACP (L'analyse en composantes principales) :.....	10
3.2.3.	Le descripteur HOG (Histogrammes de gradients orientés)	13
3.3.	Classification	15
3.3.1.	Les réseaux de neurones.....	15
3.3.2.	Le perceptron multicouche (MLP).....	17
3.3.3.	Réseau de neurone convolutionnel.....	22
4.	Diagramme de l'application générale de notre application	25
1.	Introduction.....	27
2.	Ressources utilisées	27
3.	La base de données :	27
4.	L'extraction des descripteurs	28
5.	Classification des descripteurs.....	29
6.	Résultats obtenus	29
6.1.	Approche 1 : classification avec MLP.....	29
6.1.1.	MLP avec une image brute.....	29
6.2.	Approche 2 : classification avec CNN	32
6.3.	Comparaison entre MLP et CNN	32
7.	Validation.....	33
8.	Conclusion	37
	Conclusion générale	39
	Références Bibliographiques.....	40
	Liste des Références webographies.....	41

Liste des tableaux

Tableau 1 : Présentation des plus importants travaux dans le domaine de la reconnaissance des panneaux routiers	7
Tableau 2 : principale fonction d'activation.....	18
Tableau 3 : Taux de reconnaissance (%) MLP avec une image brute.....	29
Tableau 4 : Taux de reconnaissance (%) MLP avec un prétraitement LBP	30
Tableau 5 : Taux de reconnaissance (%) MLP avec prétraitement ACP	30
Tableau 6 : Taux de reconnaissance (%) MLP avec un prétraitement HOG.....	31
Tableau 7 : Taux de reconnaissance (%) et temps d'exécution de l'algorithme CNN en fonction du nombre d'itérations.....	32
Tableau 8 : les panneaux routiers de la base de données GTSRB.....	36
Tableau 9 : Matrice de confusion MLP_LBP.....	Error! Bookmark not defined.

Liste des figures

Figure 1 : Les différentes modalités de reconnaissance : a) ADN, b) Oreille, c) visage, d) visage infrarouge, e) thermogramme main, f) veine main, g) Empreintes digitales, h) marche, i) geste j) iris, k) empreinte de la paume, l) rétine, m) signature, n) voix.	4
Figure 2 : Schéma général d'un système de reconnaissance des formes.	5
Figure 3 : Le système détecte une vitesse limitée à 80 km/h et l'affiche.....	8
Figure 4 : Schéma général de la reconnaissance des panneaux routiers.....	8
Figure 5 : Local binaire modèle (LBP) de calcul. (A) Exemple de l'opérateur LBP de base. (B) Exemples de voisinages LBP circulaires courantes : (8,1), (8,2) et (16,2) respectivement [13].....	9
Figure 6 : Exemple des données d'ACP.....	10
Figure 7 : Représenter graphiques des individus.....	13
Figure 8 : Principe de décomposition pour le calcul des HOG	13
Figure 9 : Image d'un piéton avec un gradient horizontal et vertical.....	14
Figure 10 : Image d'un piéton avec une orientation et norme du gradient.....	14
Figure 11 : Découpage d'une image en 10*6 cellules.....	15
Figure 12: Neurone formel et neurone biologique	17
Figure 13 : Structure d'un MLP	17
Figure 14 : Exemple réseaux neurone multicouche.....	20
Figure 15 : Propagation avant.....	20
Figure 16 : Rétropropagation	21
Figure 17 : Le réseau après la mise à jour des poids	22
Figure 18 : L'architecture d'une CNN	23
Figure 19 : Convoluer l'image avec un filtre de cercle	24
Figure 20 : Diagramme général de notre application	26
Figure 21 : Les panneaux routiers GTSRB	28

Liste d'abréviations

RDF : Reconnaissance des Formes.

TSR : Traffic Sign Recognition, Reconnaissance de Panneaux routiers.

LBP : local Binary Pattern, motifs binaires locaux.

ACP : Analyse en Composantes principales.

HOG : Histogram of Oriented Gradients, Histogramme de Gradients Orientés.

MLP : Multi Layer Perceptron, réseaux de neurone multicouche.

CNN : Convolutional Neural Networks, Réseaux de neurones convolutionnels.

GTSRB: German Traffic Sign Recognition Benchmark

Introduction générale

La vision est un système complexe qui peut être comparé très simplement à un appareil photo.

La pupille au centre de l'œil joue le rôle de diaphragme et le cristallin fait la mise au point exactement comme un objectif d'appareil photo. Les images se projettent sur la pellicule photo, pour l'œil c'est la rétine, située au fond de l'œil. Pour voir, l'œil transmet au cerveau les informations lumineuses qu'il reçoit. En effet, la rétine transforme la lumière reçue en impulsions électriques que le cerveau traduit en images : c'est le phénomène de vision. C'est donc le cerveau qui délivre la vision, l'œil et ses annexes sont des organes intermédiaires permettant d'accéder au résultat final : voir.

Ainsi ce parallèle avec le fonctionnement de la vision, nous amène à notre sujet principal qui est la reconnaissance des formes et des panneaux de signalisation routiers, car outre la possibilité de pouvoir voir les formes et les panneaux, nous avons besoin de systèmes capables de les reconnaître, de les traiter et de les identifier.

Les panneaux de signalisation routière sont des amers visuels cruciaux pour la navigation dans nos environnements urbains. La détection et la reconnaissance des panneaux dans des images sont très utiles pour peupler et enrichir des bases de données de routes

avec le système de reconnaissance des panneaux de signalisation, aucune interdiction de doubler et aucune limitation de vitesse n'en pourront plus vous échapper. Une caméra multifonction dotée d'une fonction de reconnaissance est pointée sur la zone située devant le véhicule. Elle détecte les panneaux de signalisation et les compare aux modèles enregistrés dans une base de données. Le signal de limitation de vitesse ou d'interdiction de doubler est affiché sur le tableau de bord, vous évitant ainsi de désagréables conséquences.

Le système reconnaît les panneaux de signalisation situés au-dessus de la chaussée ou sur les côtés droit et gauche de celle-ci ainsi que les panneaux à affichage variable (panneaux lumineux).

Le système reconnaît également les panneaux supplémentaires par exemple : par temps de pluie, camions, motos, flèches indiquant le sens de circulation.

Donc la technologie de reconnaissance des panneaux de signalisation permet de lire les panneaux permanents et temporaires situés au bord ou au-dessus de la route. Un panneau de limite de vitesse ou d'interdiction de dépasser vous est signalé par une icône sur le tableau de bord et change à chaque fois que vous entrez dans une zone où s'appliquent de nouvelles limites.

Ce travail sur la reconnaissance des panneaux de signalisation routiers a déjà été repris par l'industrie automobile qui s'est emparée du sujet dans le cadre de la conduite assistée par ordinateur. Aujourd'hui nombre de grands noms de l'automobile ont intégré dans leur véhicule des aides à la conduite grâce à la caméra de bord (détection de piétons, mesure des écarts de trajectoire, gestion des feux de route), cette dernière peut aussi être utilisée pour scanner les panneaux de signalisation.

La fonction a pour nom TSR (Traffic Sign Recognition). Le capteur vidéo détecte les limitations de vitesse et les affiches à la volée sur le tableau de bord quand on passe devant.

C'est une aide utile pour le conducteur qui, dans le trafic ambiant, n'aura pas forcément vu le panneau. Selon les systèmes, la voiture va jusqu'à comparer sa vitesse par rapport à la limitation en vigueur et alerter le conducteur s'il est en survitesse.

Ce rapport est composé de 3 chapitres :

Le chapitre 1 nous présente la reconnaissance des formes, le chapitre 2 donne une description détaillée du schéma général de la reconnaissance des panneaux routiers et le chapitre 3 décrit les outils mis en place pour la conception et l'implémentation de nos tests.

1. Introduction

La reconnaissance des formes (ou RDF) est issue de différentes disciplines qui sont les mathématiques (probabilistes et statistiques), les sciences de l'ingénieur, l'informatique et l'intelligence artificielle. C'est à partir des années 60 que la reconnaissance des formes est devenue une discipline spécifique [01]. L'extraordinaire développement des ordinateurs ces dernières années a donné un élan à la RdF en permettant des applications temps réel, en particulier dans le domaine des applications visuelles et auditives. Les procédés d'acquisition tels que camera, scanner sont très accessibles, ainsi que des ordinateurs à la fois puissants et bon marché. Ils permettent le traitement de nombreuses données en un temps raisonnable comme cela est souvent nécessaire en RdF.

2. Définition de la reconnaissance des formes

La reconnaissance des formes (RdF) -en anglais pattern recognition ou reconnaissance de motifs est un sous-domaine de l'apprentissage automatique. Elle consiste en une automatisation de tâches de perception artificielle par exemple : reconnaître un caractère, un son, un signal, un objet dans une image numérique. Elle consiste à attribuer automatiquement une étiquette à une forme présentée dans une image.

3. Domaines d'application de la reconnaissance des formes

Nous pouvons citer certains de ces domaines d'application :

3.1. Reconnaissance des formes sur signaux temporels

La parole :

La compréhension de la parole peut être effectuée quel que soit le locuteur et quelle que soit la langue. Cette discipline comprend plusieurs axes de recherche : La reconnaissance de phonèmes, la reconnaissance de mots isolés, la reconnaissance de locuteur, et le plus difficile la compréhension du discours continu.

Parmi les recherches et les applications courantes : La commande vocale, la bureautique pour la saisie et l'archivage de discours, etc. La médecine par l'analyse des signaux d'électrocardiogramme, par l'échographie, le diagnostic médical.

3.2. Reconnaissance des formes dans les images numériques

La vision : La vision comprend le traitement, l'analyse et l'interprétation des images, on y trouve :

Identification des personnes à partir des empreintes digitales, de la forme des mains, des empreintes rétiniennes, etc. (par exemple pour l'accès à des lieux sécurisés). L'analyse de radiographies, la reconnaissance de chromosomes et le comptage de globules l'identification en milieu industriel, comme la détection de défauts dans les pièces. L'analyse des images satellites (météorologies, agriculture, analyse de sol, surveillance militaire). L'analyse d'écho radar (pour la poursuite de cibles et le pilotage de missiles). L'analyse de scènes pour le déplacement d'un robot.

La lecture de l'écriture manuscrite et d'imprimerie :

Cette discipline comprend la reconnaissance des caractères isolés, la reconnaissance de mots et de phrases, la reconnaissance du script. Le tri postal et la lecture des adresses, la reconnaissance de signatures en milieu bancaire et commercial.

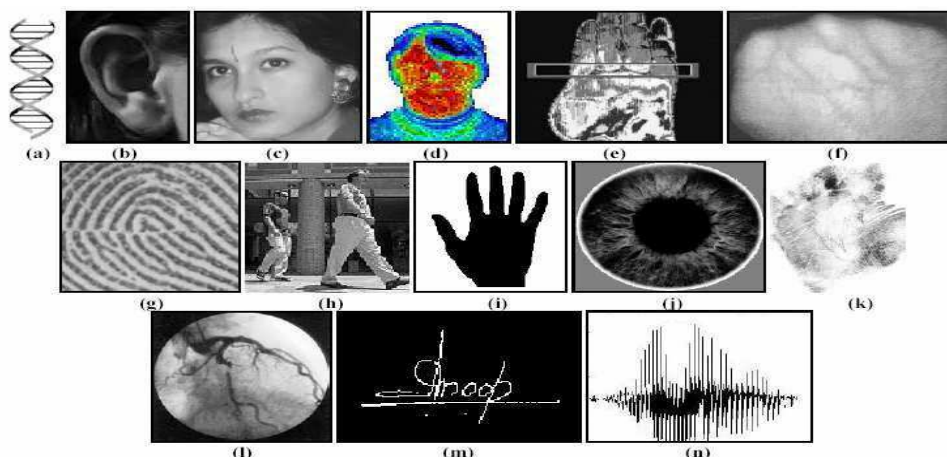


Figure 1: Les différentes modalités de reconnaissance : a) ADN, b) Oreille, c) visage, d) visage infrarouge, e) thermo gramme main, f) veine main, g) Empreintes digitales, h) marche, i) geste j) iris, k) empreinte de la paume, l) rétine, m) signature, n) voix.

4. Les Méthodes de la Reconnaissance des Formes

Il n'y a pas une théorie unifiée de RdF cependant, il existe deux approches principales, l'approche basée sur la théorie statistique de la décision (Approche par partition d'espace) et l'approche structurelle.

4.1. Approche statistique :

Des approches classiques en reconnaissance des formes sont fondées sur l'étude statistique mesures que l'on a effectuées sur les objets à reconnaître. L'étude de leur répartition dans un espace métrique et la caractérisation statistique des classes permet de prendre une décision de reconnaissance du type « plus forte probabilité d'appartenance à une classe ». Ces méthodes s'appuient en général sur des familles d'objets analogues [02] dans l'espace de représentation.

4.2. Approche structurelle : (ou syntaxique)

Ces sont des approches qui sont basées sur l'extraction de primitives en prenant compte de l'information structurelle [02]. Ces approches cherchent à structurer l'information en décrivant l'organisation topologique (la structure) de la forme à partir de ses composantes les plus élémentaires. Ces approches nécessitent une mesure de la similarité entre deux représentations structurelles. On distingue plusieurs techniques telles que les structures de graphes, les structures syntaxiques et les arbres des décisions.

5. Structure générale d'un système de reconnaissance des formes

On s'appuie sur le schéma classique d'un processus de reconnaissance de formes (Figure 1.1) pour décrire les principaux traitements à effectuer et leurs objectifs.

Les systèmes de reconnaissance des formes sont généralement basés sur quatre étapes principales : acquisition et prétraitements, extraction de caractéristiques, classification et Décision.

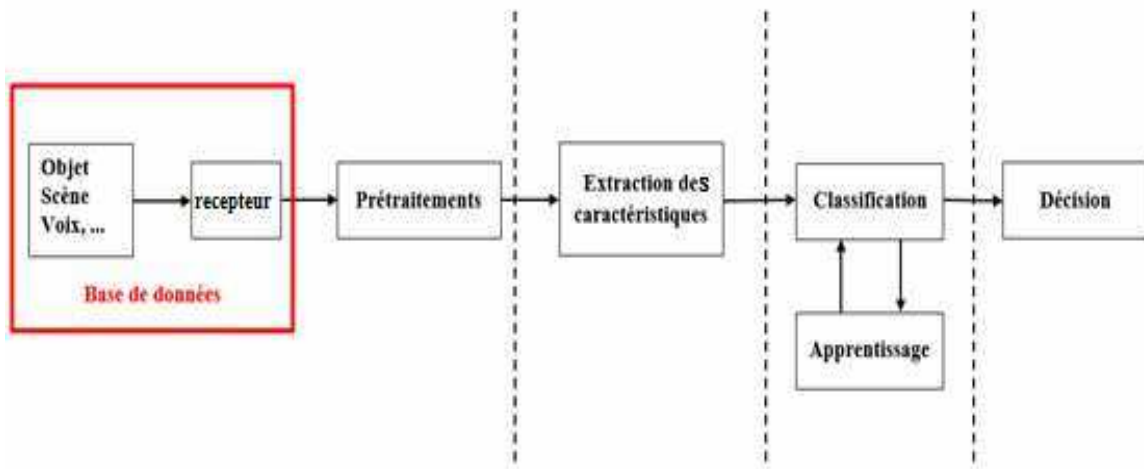


Figure 2: Schéma général d'un système de reconnaissance des formes. [01]

5.1. Acquisition et prétraitements

Suivant la nature du signal, un capteur (caméra, microphone ...) est nécessaire pour acquérir le signal sous forme numérique ou analogique (il faut alors le convertir en numérique) pour qu'il soit traitable par un système informatisé

Les images originales doivent être prétraitées afin de fournir la forme adéquate à la reconnaissance. Les principaux objectifs du prétraitement sont : la Binarisation, la réduction de bruit.

5.2. Extraction de caractéristiques

On appelle caractéristique (ou descripteur) une information qui peut être mesurée sur la donnée à reconnaître. Par exemple : l'amplitude moyenne d'un signal sur une fenêtre temporelle, l'énergie dans une bande de fréquence, le rapport hauteur sur largeur d'un caractère manuscrit ou le niveau de gris moyen d'une zone d'image, etc.

5.3. La classification

Cette phase est le noyau de la Reconnaissance des formes. En utilisant les modèles (paramètres) obtenus lors de l'apprentissage, le but de la classification est de déterminer le type (ou classe) d'un Objet.

5.4. Apprentissage

L'apprentissage ou entraînement est une partie importante du système de reconnaissance. Le classificateur étant généralement une fonction paramétrique, l'apprentissage va permettre d'optimiser les paramètres du classificateur pour le problème à résoudre, en utilisant des données d'entraînement. Lorsque les données d'entraînement sont préalablement classées, l'apprentissage est dit supervisé, sinon il est non supervisé.

a) Apprentissage supervisé :

Dans le cas de l'apprentissage supervisé, un échantillon représentatif de l'ensemble des formes à reconnaître est fourni au module d'apprentissage. Chaque forme est étiquetée par un opérateur appelé professeur. Cette étiquette permet d'indiquer au module d'apprentissage la classe dans laquelle le professeur souhaite que la forme soit rangée. Cette phase d'apprentissage consiste à analyser les ressemblances entre les éléments d'une même classe et les dissemblances entre les éléments de classes différentes pour en déduire la meilleure partition de l'espace des représentations. Les paramètres décrivant cette partition sont stockés dans une table d'apprentissage à laquelle le module de décision se référera ensuite pour classer les formes qui lui sont présentées.

b) Apprentissage non supervisé :

Dans le cas de l'apprentissage non supervisé, on fournit au système de reconnaissance un grand nombre de formes non étiquetées. L'étape de la classification va se charger d'identifier automatiquement les formes appartenant à une même classe.

c) Apprentissage semi-supervisé :

L'apprentissage semi-supervisé est à mi-chemin entre les approches supervisées et non supervisées. Les classes sont connues lors de l'apprentissage, mais l'algorithme accepte en entrée des exemples qui ne sont pas forcément étiquetés

d) Apprentissage Offline ou hors-ligne :

En apprentissage offline, tous les exemples sont connus dès le début. Il n'est plus possible par la suite d'en ajouter ou d'en enlever à moins de relancer toute la procédure. Cela permet d'avoir une vision globale du problème à traiter sous réserve d'être capable de stocker toutes les données. En contrepartie, le classifieur obtenu est figé et ne peut donc pas s'adapter aux changements ou à une autre scène.

e) Apprentissage online ou en ligne :

Dans le cas online, les exemples sont obtenus les uns après les autres et servent à mettre à jour le classifieur de manière séquentielle. Une fois utilisé, un exemple est rejeté et ne sert plus pour la suite de l'apprentissage. Ces techniques peuvent traiter de grandes quantités de données, mais l'algorithme ne voit qu'une petite partie du problème à chaque fois. Il est donc plus difficile d'avoir un classifieur pertinent dans tous les cas de figure. Néanmoins, ce dernier apprend continuellement et il est capable de s'adapter à de nouvelles conditions.

f) Apprentissage actif :

L'apprentissage actif est une autre manière de traiter le cas où la base contient des données non labellisées. Dans l'ensemble d'apprentissages, toutes les observations n'apportent pas la même quantité d'information. Certaines observations sont plus importantes que d'autres, car elles sont plus proches de la frontière ou, car elles sont dans une zone avec peu d'observations...L'apprentissage actif consiste à laisser à l'algorithme le choix de déterminer les exemples les plus pertinents parmi ceux qui ne sont pas labellisés. À charge ensuite à une entité externe, un humain par exemple, d'étiqueter les observations sélectionnées.

5.5. La décision

Une meilleure classification donne des meilleures décisions.

6. Le but de la reconnaissance des formes

Les motivations qui conduisent à faire effectuer des tâches de reconnaissance par des machines sont nombreuses. L'imitation des facultés de reconnaissance de l'homme sera toujours un moteur pour le progrès dans ce domaine. Toutefois il est important de signaler que ceci doit se faire non pas pour remplacer l'homme par une machine, mais pour le soulager des tâches ennuyeuses voir dangereuses.

7. Tableau récapitulatif

Le tableau suivant présente les plus importants travaux dans le domaine de la reconnaissance des panneaux routiers :

Gouape	Méthodes	Taux de reconnaissance avec tous les signes	Taux de reconnaissance avec les signes bleu	Taux de reconnaissance avec les signes danger
IDSIA[03]	Committee of CNNs	99.46%	99.89%	99.07%
COSFIRE [04]	Color-blob-based COSFIRE filters for object recogn	98.97%	99.83%	97.96%
INI-RTCV[05]	Human Performance	98.84%	99.72%	98.67%
SEMANET[06]	Multi-Scale CNNs	98.31%	97.18%	98.03%
CAORE[07]	Random Forests	96.14%	99.27%	92.08%
INI-RTCV[08]	LDA on HOG 2	95.68%	97.18%	93.73%
INI-RTCV[8]	LDA on HOG 1	93.18%	96.10%	90.61%
INI-RTCV[8]	LDA on HOG 3	92.34%	95.20%	88.24%

Tableau 1 : Présentation des plus importants travaux dans le domaine de la reconnaissance des panneaux routiers

8. Conclusion

De manière générale, on peut dire que les systèmes de reconnaissance des formes sont des fonctions de classification qui associent à toute forme inconnue sa classe la plus probable. Malgré la difficulté de la tâche, les systèmes de reconnaissance des formes ont connu un grand succès dans des domaines variés.

Un système de reconnaissance des formes contient principalement quatre étapes. Une étape d'acquisition et prétraitement des données qui est utile pour éliminer des bruits, une étape d'extraction des descripteurs des données qui se charge de faciliter la tâche de reconnaissance en présentant les données en entrée sous une forme adaptée à la classification une étape de classification qui va classer les formes inconnues, est le dernier étape la décision.

1. Introduction :

Parmi les grands champs d'utilisation de la reconnaissance des formes la reconnaissance des panneaux routiers (TSR). La majorité des articles de reconnaissance de panneaux de signalisation se focalisent sur trois techniques principales : Détection des panneaux, Estimation de forme et reconnaissance des panneaux. Dans ce chapitre, nous abordons la deuxième principale étape d'une TSR qui est la reconnaissance des panneaux (classification). Nous commençons par présenter de manière générale les différentes étapes pour la reconnaissance de panneaux routiers.

2. Définition de la reconnaissance des panneaux (TSR) :

Comme son nom l'indique, la reconnaissance des panneaux est un équipement automobile qui lit et interprète les panneaux de signalisation sur la route, afin d'en informer le conducteur au cas où il n'aurait pu les voir. Les panneaux de limitation de [w1] vitesse et d'interdiction de dépassement sont notamment concernés.



Figure 3: Le système détecte une vitesse limitée à 80 km/h et l'affiche

Dans ce travail, nous nous intéressons à l'étape de reconnaissance. Cette dernière identifie les panneaux détectés dans les images fournies par la caméra en exploitant leurs caractéristiques visuelles. La reconnaissance nécessite trois phases : prétraitement, l'extraction des caractéristiques et la classification des descripteurs (figure 4).

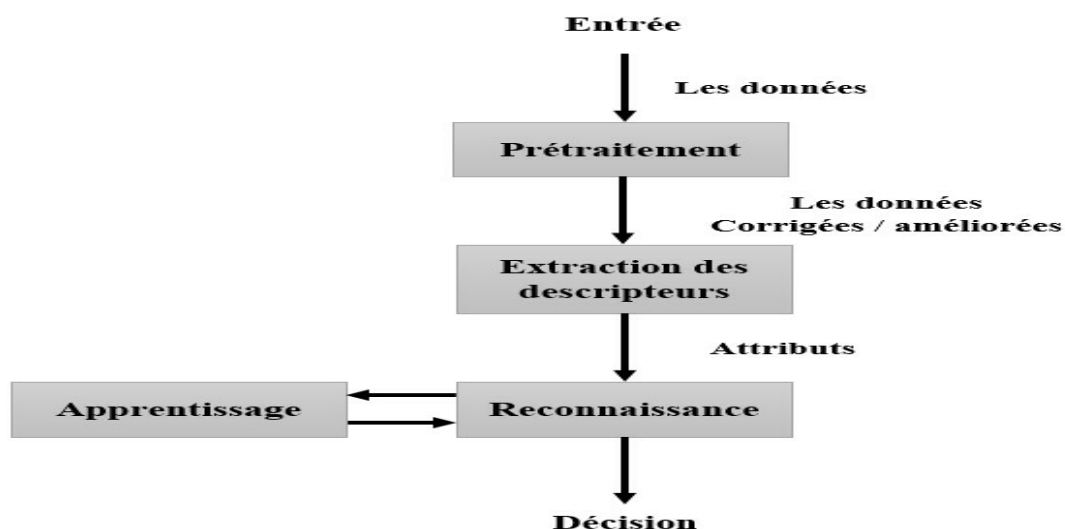


Figure 4: Schéma général de la reconnaissance des panneaux routiers

3. Les étapes de TSR

3.1. Prétraitement :

Les images originales doivent être prétraitées afin de fournir la forme adéquate à la reconnaissance. Les principaux objectifs du prétraitement sont : la réduction de bruit, la normalisation [9]. D'une façon générale, le résultat du prétraitement doit être une image contenant le champ à reconnaître sans n'importe quel autre élément ennuyeux.

3.2. Extraction des caractéristiques :

L'objectif de l'extraction des descripteurs (caractéristiques) dans le domaine de la reconnaissance des formes consiste à exprimer les primitives sous une forme numérique ou symbolique appelée codage. Dans cette partie, nous allons introduire les descripteurs qui seront utilisés dans la partie Expérimentations et résultats il s'agit tout d'abord les motifs binaires locaux(LBP), puis l'analyse en composantes principales (ACP) et enfin des histogrammes de gradients(HOG).

3.2.1. Le descripteur LBP (motifs binaires locaux) :

Les motifs binaires locaux ont initialement été proposés par Ojala en 1996 afin de caractériser les textures présentes dans des images en niveaux de gris [10]. Ils consistent attribuer à chaque pixel P de l'image I (i , j) à analyser, une valeur caractérisant le motif local autour de ce pixel. Ces valeurs sont calculées en comparant le niveau de gris du pixel central P aux valeurs des niveaux de gris des pixels voisins.

Le concept du LBP est simple, il propose d'assigner un code binaire à un pixel en fonction de son voisinage. Ce code décrivant la texture locale d'une région est calculé par seuillage d'un voisinage avec le niveau de gris du pixel central. Afin de générer un motif binaire, tous les voisins prendront alors une valeur "1" si leur valeur est supérieure ou égale au pixel courant et "0" autrement (**Figure 2.5**). Les pixels de ce motif binaire sont alors multipliés par des poids et sommés afin d'obtenir un code LBP du pixel courant. On obtient donc pour toute l'image, des pixels dont l'intensité se situe entre 0 et 255 comme dans une image à 8 bits ordinaire. Plutôt que de décrire l'image par la séquence des motifs LBP, on peut choisir comme descripteur de texture un histogramme de dimension 255.

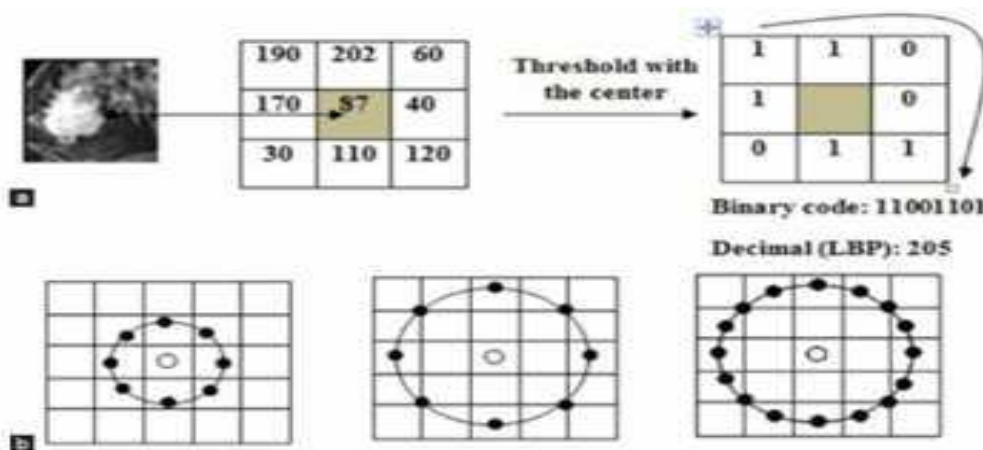


Figure 5: Local binaire modèle (LBP) de calcul. (A) Exemple de l'opérateur LBP de base. (B) Exemples de voisinages LBP circulaires courantes : (8,1) (8,2) et (16,2) respectivement [11]

soit g_c le niveau de gris du pixel central, g_p ($p = 1 \dots P$) les niveaux de gris de ses voisins, l'indice LBP du pixel courant est calculés comme :

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

Où :

- $S(x) = \begin{cases} 1 & \text{si } x \geq 0 \\ 0 & \text{si } x < 0 \end{cases}$
- (x_c, y_c) Sont les coordonnées du pixel courant.
- $LBP_{P,R}$ Est le code LBP pour le rayon R et le nombre de voisins P.

L'opérateur LBP obtenue avec $P = 8$ et $R = 1$ (LBP 8,1) est très proche de l'opérateur LBP d'origine. La principale différence est que les pixels doivent d'abord être interpolés pour obtenir les valeurs des points sur le cercle (voisinage circulaire au lieu de rectangulaire).

La propriété importante du code LBP est que ce code est invariant aux changements uniformes globaux d'illumination parce que la LBP d'un pixel ne dépend que des différences entre son niveau de gris et celui de ses voisins.

3.2.2. Le descripteur ACP (l'analyse en composantes principales) :

L'analyse en composantes principales (ACP), développée en France dans les années 1960 par JP. Benzécri est une méthode de statistique exploratoire permettant de décrire un grand tableau de données de type individus / variables.

L'ACP fait en réalité partie d'un ensemble de méthodes d'analyse de données, appelées méthodes multifactorielles. De façon générale, celles-ci ont pour but de résumer de la façon la plus fidèle possible un grand ensemble de données. Dans notre cas l'utilité de l'ACP est la réduction de l'espace de représentation.

	N1	N2	N3
I1	12	11	10
I2	11	08	13
I3	09	13	07
I4	12	11	08
I5	13	10	15
I6	19	15	10

Figure 6: Exemple des données d'ACP

- L'Algorithme d'ACP :

Algorithme : analyse en composantes principales (ACP)

Début

Centrer le tableau $\{X(n, p)\}$:

Calculer la matrice Variance Covariance (V) :

Déterminer la métrique M

Si la même unité de mesure

| Homogènes

Si non

Hétérogène

Recherche des axes principaux UK de matrice (VM)

Calculer les valeurs propres

Trier les valeurs propres par ordre décroissant : $\lambda_1 > \lambda_2 \dots > \lambda_p$

Calculer la qualité de représentation Q_j

Si $Q_j \geq 80\%$

| Oui

Si non

Non

Calculer les vecteurs u_k de matrice VM

Les composantes principales

Représenter graphiquement les individus

Fin

▪ **Algorithme détaillé :**

- Centrer le tableau $\{X(n, p)\}$:

$$X = X - g \quad (1)$$

Avec :

X : les données

g :le centre de gravité

$$g = \frac{\sum_{i=1}^n P_i X_{ik}}{\sum_{i=1}^n P_i} \quad (2)$$

- Calculer la matrice Variance Covariance (V) :

$$V = \frac{1}{N} X^t X \quad (3)$$

- Déterminer la métrique M

Dans notre cas on a la même unité de mesure « pixels » (*homogènes*) :

$$M = I = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4)$$

- Recherche des axes principaux UK de matrice (VM)
- Calculer les valeurs propres

$$\text{dét}(\mathbf{VM} - \lambda \mathbf{I}) = 0 \quad (5)$$

On obtient le polynôme et après on utilise la division euclidienne pour trouver les valeurs propres λ

- Trier les valeurs propres par ordre décroissant : $\lambda_1 > \lambda_2 \dots > \lambda_p$

$$\lambda = \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_p \end{bmatrix} \quad (6)$$

- Calculer la qualité de représentation Q_j

$$Q_j = \frac{\sum_{i=1}^j \lambda_i}{\sum_{i=1}^p \lambda_i} \geq 80\% \quad (7)$$

$$Q_1 = \frac{\lambda_1}{\lambda_1 + \lambda_2 + \lambda_3} = 54\% \text{ non}$$

$$Q_2 = \frac{\lambda_1 + \lambda_2}{\lambda_1 + \lambda_2 + \lambda_3} = 90\% \text{ oui (} \lambda_p = \lambda_1, \lambda_2 \text{)}$$

- Calculer les vecteurs propre u_k de matrice VM

$$\mathbf{VM}u_k = \lambda_p u_k \quad (8)$$

- Les composantes principales

$$C_k = \mathbf{XM}u_k \quad (9)$$

- Représenter graphiquement les individus dans l'espace réduit en utilisant les composantes principales

▪ **Exemple :**

$$C_k = \begin{bmatrix} 2\sqrt{6} & -2\sqrt{3} \\ -5\sqrt{5} & 0 \\ \sqrt{6} & 2\sqrt{3} \\ \sqrt{6} & 2\sqrt{3} \\ -\sqrt{6} & 0 \\ -2\sqrt{6} & -2\sqrt{3} \end{bmatrix} \begin{cases} I1 \\ I2 \\ I3 \\ I4 \\ I5 \\ I6 \end{cases}$$

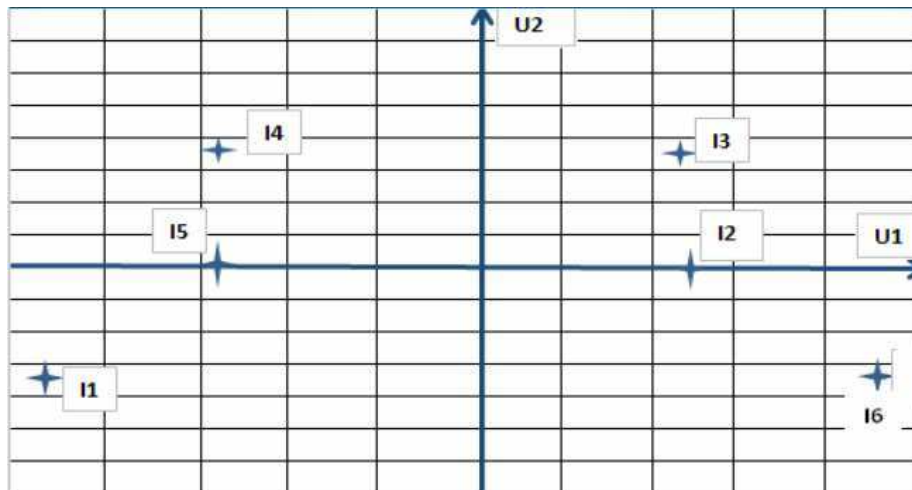


Figure 7: Représenter graphiques des individus

3.2.3. Le descripteur HOG (Histogrammes de gradients orientés)

Les histogrammes de gradients orientés (Histograms of Oriented Gradients ou HOG) et leurs variantes sont sans conteste les descripteurs les plus populaires dans le domaine de la détection de piétons.

Ils ont été introduits par Dalal et Triggs, 2005 [12] et s'inspirent fortement des travaux précédents de Lowe, 2004 [13] sur le descripteur SIFT (Scale-invariant feature transform). L'idée mise en œuvre ici est de discrétiser l'orientation du gradient dans l'image à l'aide d'histogrammes. Ainsi les contours, donc la forme des objets, sont codés ce qui permet de les reconnaître.

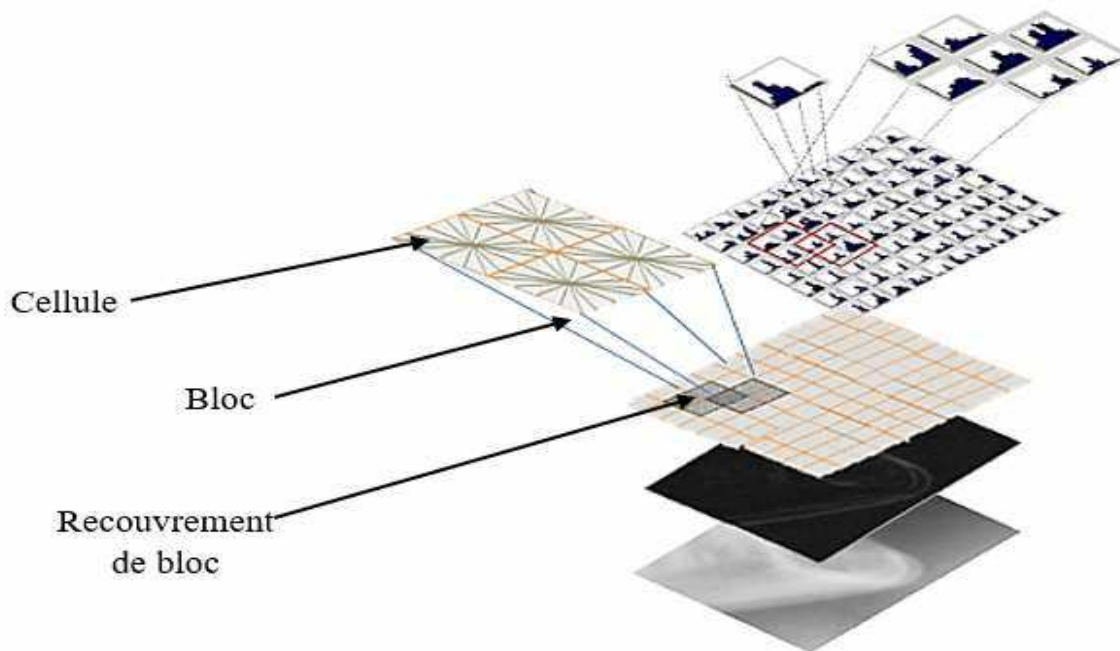


Figure 8 : Principe de décomposition pour le calcul des HOG

3.2.4. Calcul du gradient

D'un point de vue pratique, le calcul d'un HOG commence par l'évaluation des gradients horizontaux G^x et verticaux G^y Pour tous les pixels de l'image. Ce filtrage est réalisé à l'aide de deux convolutions effectuées séparément. La première utilise le masque $[-1, 0, 1]$ pour les gradients horizontaux et la seconde emploie le masque $[-1, 0, 1]^T$ Pour les gradients verticaux. Ainsi à chaque point (i, j) de l'image est associé un couple $(G_{i,j}^x, G_{i,j}^y)$

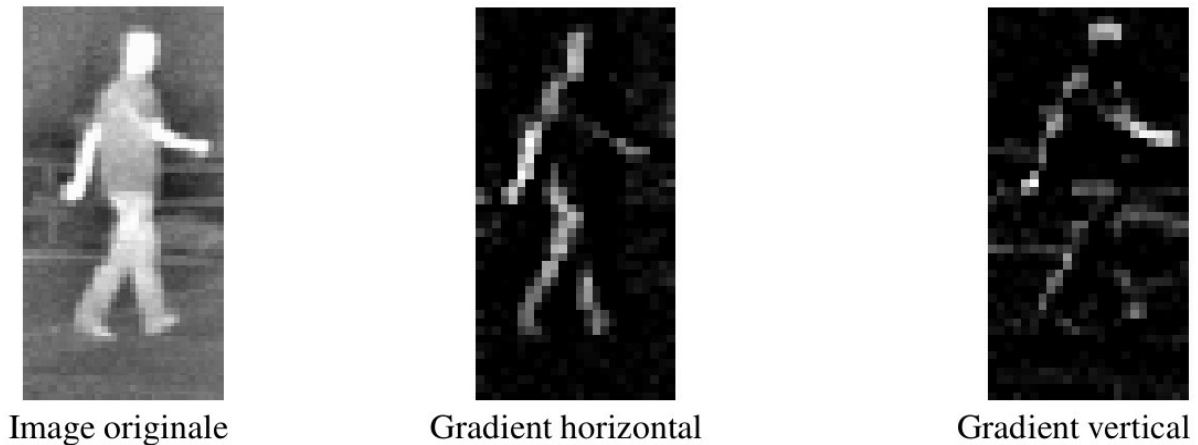


Figure 9 : Image d'un piéton avec un gradient horizontal et vertical

La norme n et l'orientation θ du gradient en un point sont alors définies par

$$n_{i,j} = \sqrt{(G_{i,j}^x)^2 + (G_{i,j}^y)^2} \quad (1)$$

$$\theta_{i,j} = \arctan\left(\frac{G_{i,j}^y}{G_{i,j}^x}\right) \quad (2)$$



Figure 10 : Image d'un piéton avec une orientation et norme du gradient

3.2.5. Découpage de l'image

L'image est ensuite décomposée en blocs, généralement de 8 pixels de côté, et un histogramme est construit pour chacun d'entre eux. Les classes des histogrammes représentent l'angle du gradient. Leur nombre est variable suivant les implémentations. Dalal et Triggs, 2005 [12] ont choisi d'en utiliser 9 dans l'intervalle $[0, \pi]$ (gradient non signé). Un pixel d'un bloc vote donc pour une classe en fonction de l'angle du gradient en ce point et son importance dans l'histogramme est la norme du gradient.

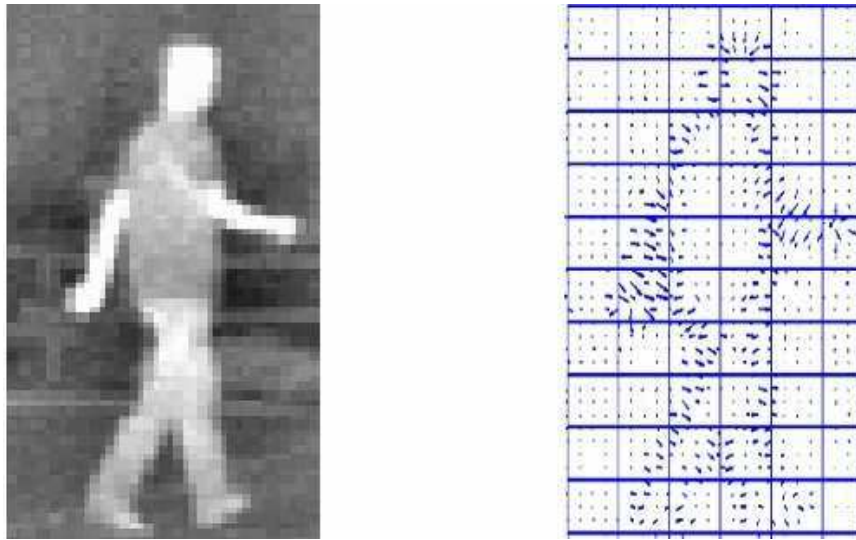


Figure 11 : Découpage d'une image en 10*6 cellules

3.2.6. Normalisation :

La dernière étape est la normalisation des histogrammes entre les blocs. Les histogrammes de blocs adjacents (par exemple des regroupements 4x4) sont concaténés pour former un vecteur qui est ensuite normalisé pour former une partie du descripteur. Le descripteur complet est obtenu en répétant l'opération sur tous les groupements de blocs possibles.

3.3. Classification

Classifier un ensemble d'objets, c'est attribuer à chacun une classe (ou une catégorie) parmi plusieurs classes définies à l'avance. Cette tâche est appelée classification ou discrimination. Un algorithme qui réalise automatiquement une classification est appelé classifieur. Pour la reconnaissance des formes, généralement, les classifieurs utilisés sont les réseaux de neurones (MLP), les réseaux bayésiens, séparateurs à Vastes Marges (SVM), dans notre travail on a réseaux de neurones multicouches (MLP) et réseaux de neurones convolutionnel (CNN), en détaillons leur principe.

3.3.1. Les réseaux de neurones

Un réseau de neurones est un outil puissant de la modélisation des relations complexes entre les données d'entrée et de sortie. Le développement de cette technologie a été motivé par l'ambition d'avoir un système artificiel capable d'accomplir des tâches "intelligentes" de manière semblable à celles qui sont exécutées par le cerveau humain. Un réseau de neurones ressemble au cerveau humain dans le sens où il apprend des connaissances par apprentissage,

et ses connaissances sont stockées dans des connexions interneurones connus sous le nom « poids synaptiques ». [14] L'avantage des réseaux de neurones se situe dans leur capacité d'apprendre à résoudre des problèmes complexes à partir la modélisation des exemples d'apprentissage réels.

Les réseaux de neurones, avec leur capacité remarquable de dériver la signification des données incompréhensibles ou imprécises, peuvent reconnaître et détecter des formes complexes qui sont difficiles à détecter par d'autres techniques. Un réseau de neurones entraîné peut être vu comme un "expert" capable de catégoriser l'information à analyser et de fournir des réponses à des nouvelles informations ou situations.

Nous présentons quelques modèles d'architecture des réseaux de neurones artificiels et leur principe de fonctionnement notamment les réseaux de neurones multicouches(MLP) et les réseaux de neurones convolutionnels(CNN).

a) Définition

Les réseaux de neurones sont des algorithmes d'apprentissage et d'optimisation basés sur des concepts inspirés du fonctionnement d'un cerveau humain. [15] Ils se sont généralement constitué des cinq composants suivants :

- 1) Un graphe dirigé qui représente la topologie du réseau.
- 2) Une variable d'état associé à chaque nœud.
- 3) Un poids associé à chaque connexion.
- 4) Un biais associé à chaque nœud.
- 5) Une fonction de transfert pour chaque nœud qui détermine l'état d'un nœud en fonction des poids de ses liens entrants, son biais et les états des nœuds reliés à ce nœud. Cette fonction est habituellement une fonction sigmoïde.

b) Neurone biologique

Un neurone biologique est constitué d'un :

- **Le stroma** : contiens le noyau
- **Les dendrites** : forme une arborisation autour du soma.
- **L'axone** : prolongation au bout duquel on trouve une arborisation qui entre en contact avec les dendrites d'autres neurones via les boutons synaptiques.
- **La synapse** : est une jonction entre deux neurones et généralement entre l'axone d'un neurone et la dendrite d'un autre

c) Neurone formel

Le neurone formel est une abstraction de la réalité biologique. Est une unité élémentaire qui reçoit un nombre d'entrées ou des sorties en provenance des autres neurones du réseau, un poids est associé à chacune de ces entrées qui représentent la puissance de sa connexion avec un neurone.

La sortie du neurone est une somme pondérée, de ses entrées plus un biais, propagé par une fonction d'activation qui peut être (sigmoïde, Heaviside...etc.).

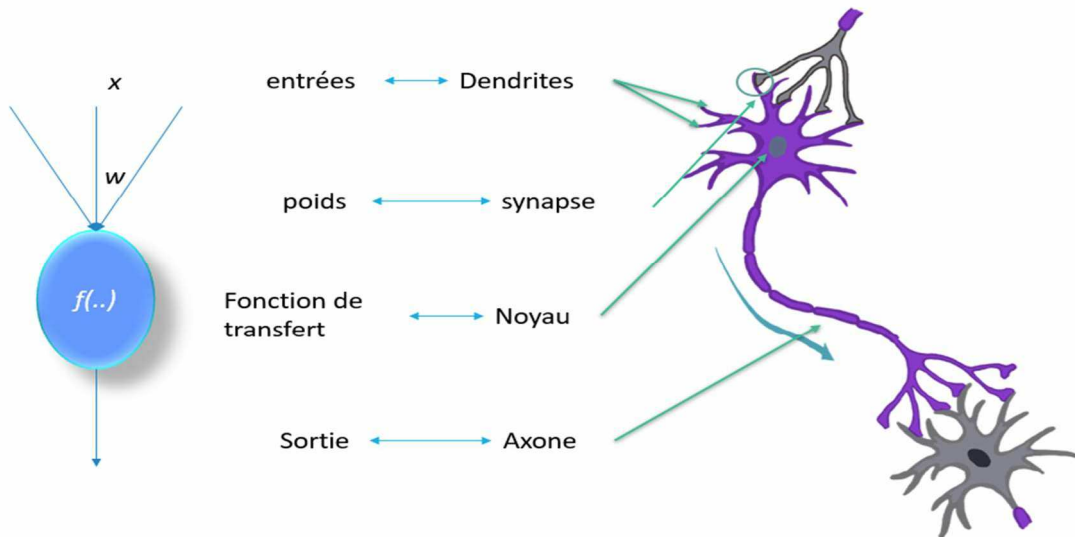


Figure 12: Neurone formel et neurone biologique

3.3.2. Le perceptron multicouche (MLP)

La perception multicouche (MLP) est l'exemple le plus répandu des classificateurs neuronaux employés dans le domaine de la reconnaissance. Un MLP est composé d'une couche d'entrée dont la taille est égale à la taille des données d'entrée, une ou plusieurs couches cachées dont la taille est déterminée par essai et la couche de sortie dont la taille est égale au nombre de classes du problème à résoudre (**Figure 13**).

Dans les réseaux MLP, chaque neurone est relié à un certain nombre d'entrées qui peuvent être les données d'entrée ou les sorties des couches précédentes chacune de ces entrées sont pondérées par un poids synaptique ; le poids total d'un neurone est le produit scalaire entre les entrées et leurs poids correspondant avec addition d'un biais (b) : Le Poids total = $X_1 W_1 + \dots + X_N W_N + b$

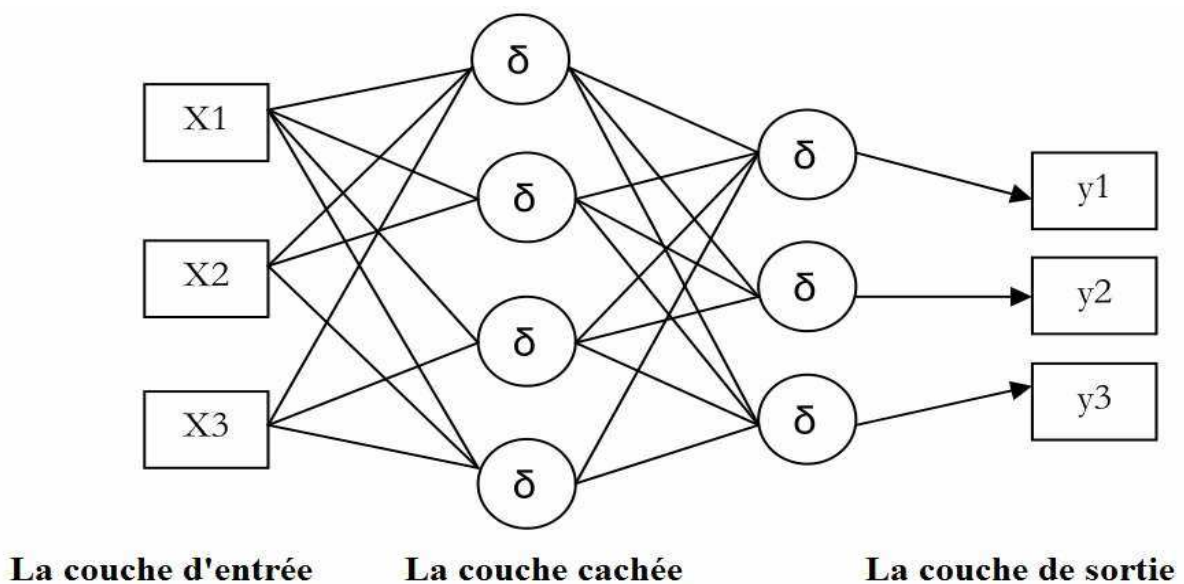


Figure 13 : Structure d'un MLP

Avec δ est une fonction d'activation (**tab 2**) employée pour propager l'information à la couche suivante.

Un algorithme d'apprentissage tel que la rétropropagation de gradient est appliqué pour ajuster les poids en réduisant au minimum une fonction d'erreur qui est, en général, la somme des erreurs quadratiques entre la sortie du réseau de neurones et la sortie attendue.

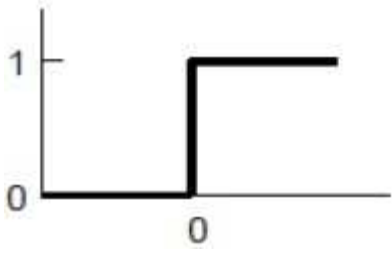
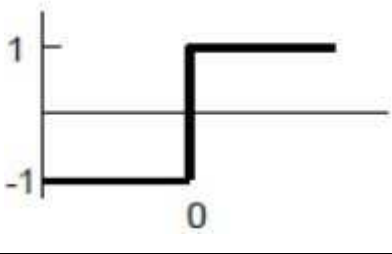
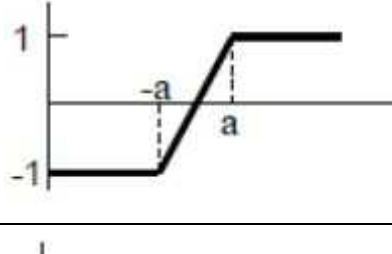
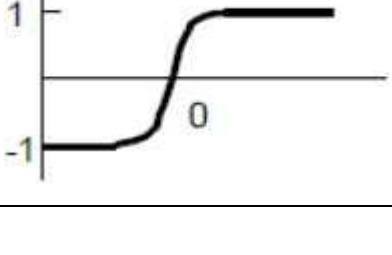
	Fonctions	Formules	Graphes
Valeurs binaires	Fonction Heaviside	$f(\sigma) = H(\sigma) = \begin{cases} 1 & \text{si } \sigma \geq 0 \\ 0 & \text{si } \sigma < 0 \end{cases}$	
	Fonction signe	$f(\sigma) = S(\sigma) = \begin{cases} 1 & \text{si } \sigma \geq 0 \\ -1 & \text{si } \sigma < 0 \end{cases}$	
	Fonction linéaire à seuil	$f(\sigma) = H(\sigma) = \begin{cases} 1 & \text{si } \sigma > 0 \\ \frac{1}{\alpha} \sigma & \text{si } \sigma \in [-\alpha, \alpha] \\ -1 & \text{si } \sigma < -\alpha \end{cases}$	
Valeurs réelles	Fonction sigmoïde	$f(\sigma) = \text{sigm}(\sigma) = \frac{\sigma}{1 + \sigma }$	

Tableau 2 : principale fonction d'activation

3.3.2.1. Le rétro propagation par descente de gradient :

Cet algorithme évalue l'erreur de gradient pour chaque neurone dans le réseau depuis la dernière couche jusqu'à la première. Les poids produisant une erreur significative seront modifiés en grande valeur par rapport aux poids produisant une erreur marginale. La rétro propagation par descente de gradient a été efficacement employée pour l'apprentissage d'un MLP ; son principe est de converger de manière itérative vers un ensemble optimal des poids synaptiques. Dans le cas de l'apprentissage supervisé, les sorties sont fournies à l'avance ; la

différence entre ces sorties et les sorties du réseau (après propagation de l'information par une fonction d'activation) constitue l'erreur à réduire au minimum par rétropropagation tout en ajustant les poids [16].

3.3.2.2 L'algorithme la rétro propagation par descente de gradient

Algorithme : La rétro propagation

Entrées : exemples, un ensemble d'exemples, chacun avec vecteur d'entrée x et vecteur de sortie y , réseaux multicouches avec couches L , poids $w_{i,j}$, fonction d'activation g

Variables locales Δ , vecteur d'erreurs indexées par nœud de réseau

Pour chaque poids w_{ij} dans le réseau, faire

$$w_{ij} \leftarrow a \text{ Petit nombre aléatoire}$$

Répéter

Pour chaque exemple (x, y) dans les exemples faire

/* Propagation des entrées vers l'avant pour calculer les sorties*/

Pour chaque nœud i dans la couche d'entrée faire

$$a_i \leftarrow x_i$$

Pour $l = 2$ à L faire a_i

$$a_j \leftarrow g(in_j)$$

Fin pour

Fin pour

/* Deltas de propagation en arrière de la couche de sortie à la couche d'entrée */

Pour chaque nœud j dans la couche de sortie faire

$$\Delta[j] \leftarrow y_i - a_i = \left(-\frac{\partial \text{loss}}{\partial in_j}\right)$$

Fin pour

Pour $l = L-1$ à 1 faire

Pour chaque nœud i en couche l faire

$$\Delta[i] \leftarrow g'(in_i)(1 - g(in_i)) \sum_j w_{ij} \Delta[j]$$

Fin pour

Fin pour

/* Mettre à jour tous les poids dans le réseau en utilisant deltas */

Pour chaque poids w_{ij} en réseau faire

$$w_{ij} \leftarrow w_{ij} + a_i * \Delta[j]$$

Fin pour

jusqu'à ce qu'un critère d'arrêt soit satisfait

Retour Réseau

3.3.2.4 Exemple détaillé du MLP a rétro propagation :

$X = [2, -1], y = 1$

Et la fonction d'activation sigmoïde : $g(x) = \frac{1}{1+e^{-x}}$

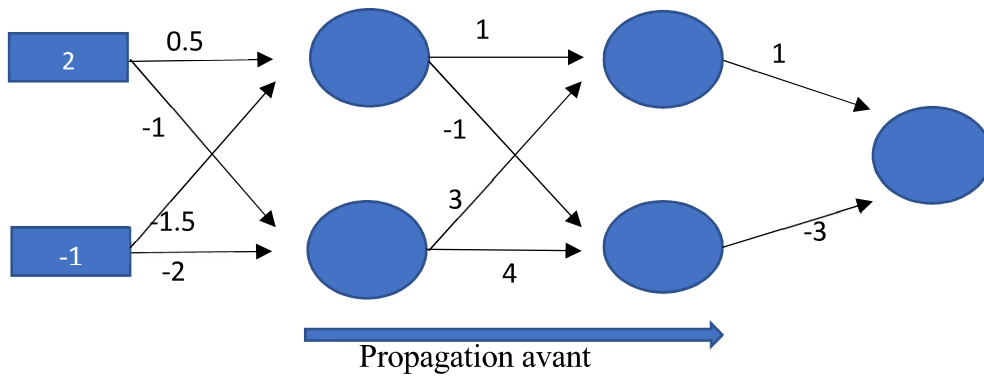


Figure 14 : Exemple de réseaux neurone multicouche

a) Partie 1 : propagation avant

Calculer l'activation de chaque neurone

$$ak = g\left(\sum_i j.k a_j\right)$$

Sigmoïde $(0.5*2+1.5*-1) = \text{sigmoïde}(-0.5) = 0.378$

Sigmoïde $(-1*2+ -2*-1) = \text{sigmoïde}(0) = 0.5$

Sigmoïde $(1*0.378+ 3*0.5) = \text{sigmoïde}(1.878) = 0.867$

Sigmoïde $(-1*0.378+ -4*0.5) = \text{sigmoïde}(-2.378) = 0.085$

Sigmoïde $(1*0.867+ -3*0.085) = \text{sigmoïde}(0.612) = 0.648$

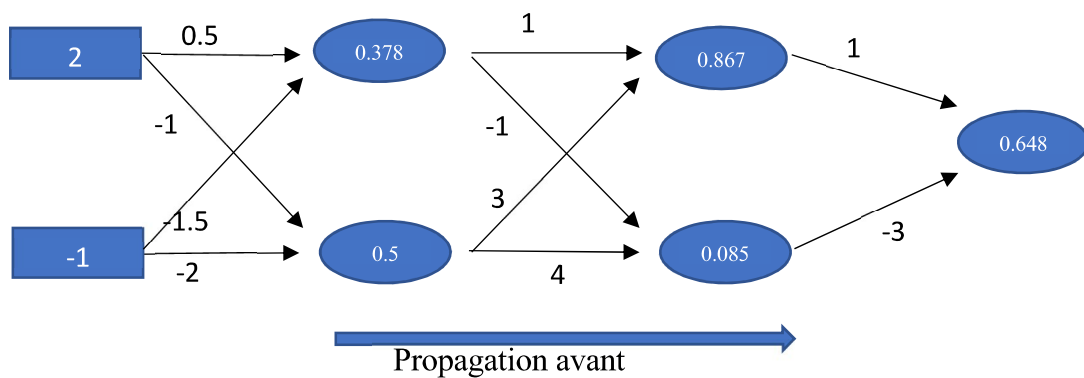


Figure 15 : Propagation avant

b) Partie 2 : rétro propagation

Calculer deltas les propagé de couche de sortie jusqu'à la couche d'entrée

$$\Delta[j] \leftarrow g(in_j)(1 - g(in_j)) \sum_k w_{j,k} \Delta[k]$$

$$\Delta_1 = 1 - 0.648 = 0.352$$

$$\Delta_2 = 0.867 * (1 - 0.867) * 1 * 0.352 = 0.041$$

$$\Delta_3 = 0.085 * (1 - 0.085) * -3 * 0.352 = -0.082$$

$$\Delta_4 = 0.378 * (1 - 0.378) * (1 * 0.041 + -1 * -0.082) = 0.029$$

$$\Delta_5 = 0.5 * (1 - 0.5) * (3 * 0.041 + -4 * -0.082) = 0.113$$

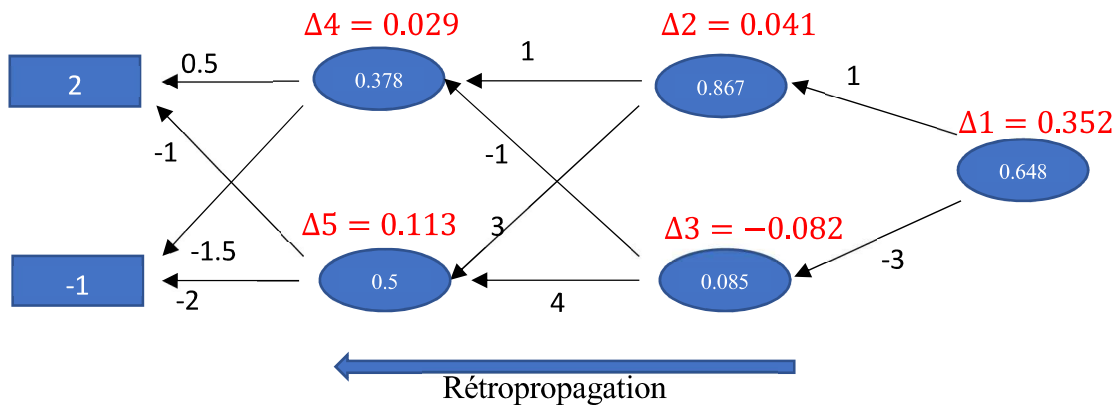


Figure 16 : Rétro propagation

c) Partie 3 : Mise a jours des pois w :

Mise à jour ($\alpha=0.1$)

$$w_{ij} \leftarrow w_{ij} + \alpha * a_i * \Delta[j]$$

$$w_{1,1} \leftarrow 0.5 + 0.1 * 2 * 0.029 = 0.506$$

$$w_{1,2} \leftarrow -1 + 0.1 * 2 * 0.113 = -0.977$$

$$w_{2,1} \leftarrow 1.5 + 0.1 * -1 * 0.029 = 1.497$$

$$w_{2,2} \leftarrow -2 + 0.1 * -1 * 0.113 = -2.011$$

$$w_{3,1} \leftarrow 1 + 0.1 * 0.378 * 0.041 = 1.002$$

$$w_{3,2} \leftarrow -1 + 0.1 * 0.378 * -0.082 = -1.003$$

$$w_{4,1} \leftarrow 3 + 0.1 * 0.5 * 0.041 = 3.002$$

$$w_{4,2} \leftarrow -4 + 0.1 * 0.5 * -0.082 = -4.004$$

$$w_{5,1} \leftarrow 1 + 0.1 * 0.867 * 0.352 = 1.031$$

$$w_{5,2} \leftarrow -3 + 0.1 * 0.085 * 0.352 = -2.997$$

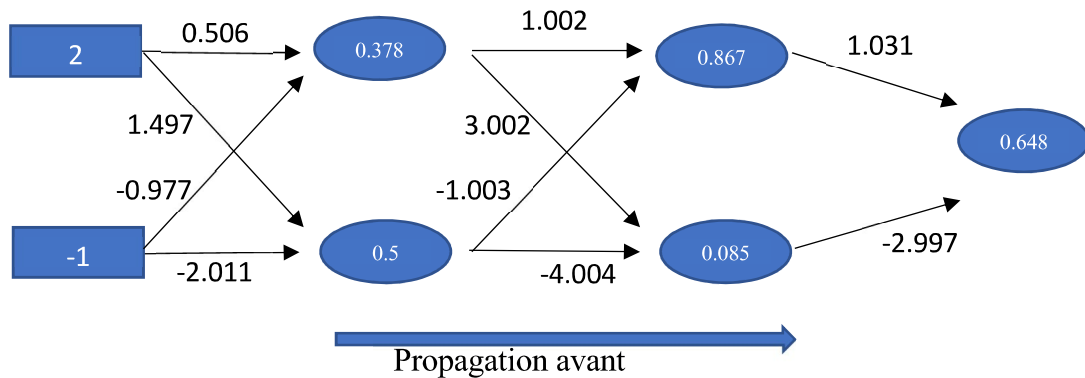


Figure 17 : Le réseau après la mise à jour des poids

3.3.2.5 Les avantages et les inconvénients

Les avantages de tels systèmes sont leur simplicité et leur rapidité de traitement. Le réseau est qualifié de "boîte noire", la classe d'appartenance de la donnée d'entrée étant directement calculée par celui-ci. L'inconvénient est que ce résultat, de fait, n'est plus interprétable par l'utilisateur. De plus, la rapidité du système est contrebalancée par la durée de la phase d'apprentissage qui peut être longue. Même si les réseaux neuronaux sont historiquement plus anciens que d'autres méthodes d'apprentissage comme les SVMs (Support Vector Machines - Séparateurs à Vaste Marge) et leur processus d'apprentissage moins déterministe et plus délicat, ils présentent l'avantage de permettre de traiter très facilement des problèmes multi classes (en mettant par exemple un neurone de sorti par classe). C'est l'une des raisons, ajoutées à leur simplicité, pour laquelle ils demeurent un outil apprécié dans les domaines impliquant des systèmes intelligents, comme les systèmes d'aide à la conduite. Dans le cadre de la détection de panneaux, de nombreuses études mettent en œuvre des réseaux de neurones pour la phase de classification [17], [18].

3.3.3. Réseau de neurones convolution

3.3.3.1 Historique

Les réseaux de neurones, comme son nom l'indique, sont une technique d'apprentissage par machine qui est inspirée de la structure du cerveau. Il comprend un réseau d'unités d'apprentissage appelées neurones. Ces neurones apprennent à convertir les signaux d'entrée (par exemple, l'image d'un chat) en signaux de sortie correspondants (par exemple, l'étiquette "chat"), formant la base de la reconnaissance automatisée.

L'intelligence des réseaux neuronaux est étrange. Alors que les réseaux neuronaux artificiels ont été étudiés au début des années 1960 par Rosenblatt, ce n'est qu'à la fin des années 2000 que l'apprentissage en profondeur utilisant les réseaux de neurones a décollé. L'outil clé a été l'échelle du pouvoir de calcul et des jeux de données avec la recherche pionnière de Google sur l'apprentissage en profondeur. En juillet 2012, les chercheurs de Google ont exposé un réseau de neurones avancé à une série d'images statiques non marquées découpées dans des

vidéos YouTube. À leur grande surprise, ils ont découvert que le réseau neuronal a eu recours à un neurone détecteur de chat seul, soutenant l'affirmation populaire selon laquelle « Internet est constitué de chats ».

La technique que les chercheurs de Google utilisent s'appelle Convolutional Neural Networks (CNN), un type de réseau neuronal artificiel avancé. Il diffère des réseaux neuronaux réguliers en termes de flux de signaux entre les neurones. Les réseaux neuronaux typiques transmettent des signaux le long du canal d'entrée-sortie dans une direction unique, sans permettre aux signaux de retourner dans le réseau. C'est ce qu'on appelle un flux vers l'avant.

Alors que les réseaux d'alimentation directe étaient utilisés avec succès pour la reconnaissance de l'image et du texte, il fallait connecter tous les neurones, ce qui entraînait une structure de réseau trop complexe. Le coût de la complexité augmente lorsque le réseau doit être formé sur de grands ensembles de données qui, conjugués aux limitations de la vitesse de traitement de l'ordinateur, entraînent des temps de formation excessivement longs. Par conséquent, les réseaux d'alimentation directe sont tombés en désuétude de l'apprentissage par machine traditionnel dans la haute résolution actuelle, la bande passante élevée, l'âge des médias. Une nouvelle solution était nécessaire.

3.3.3.2 L'architecture d'une CNN

Les réseaux de neurones convolutifs sont à ce jour les modèles les plus performants pour classer des images. Désignés par l'acronyme CNN, de l'anglais Convolutional Neural Network, ils comportent deux parties bien distinctes [w2]. En entrée, une image est fournie sous la forme d'une matrice de pixels. Elle a 2 dimensions pour une image en niveaux de gris. La couleur est représentée par une troisième dimension, de profondeur 3 pour représenter les couleurs fondamentales [Rouge, Vert, Bleu].

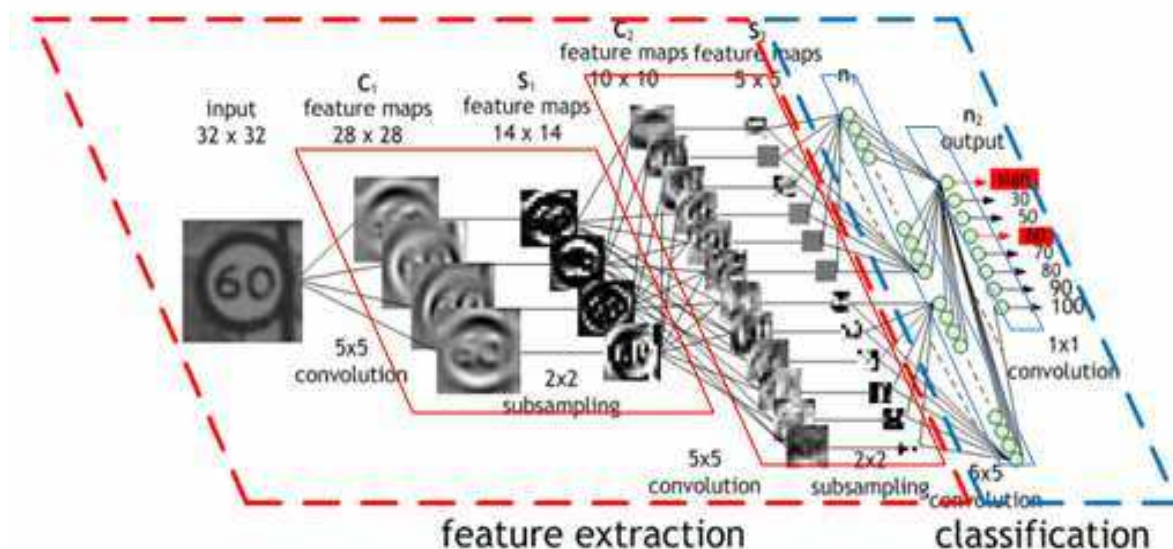


Figure 18 : L'architecture d'une CNN

La première partie d'une CNN est la partie convolutive à proprement parler. Elle fonctionne comme un extracteur de caractéristiques des images. Une image est passée à travers

une succession de filtres, ou noyaux de convolution, créant de nouvelles images appelées cartes de convolutions. Certains filtres intermédiaires réduisent la résolution de l'image par une opération de maximum local. Au final, les cartes de convolutions sont mises à plat et concaténées en un vecteur de caractéristiques, appelé code CNN.

Ce code CNN en sortie de la partie convolutive est ensuite branché en entrée d'une deuxième partie, constituée de couches entièrement connectées (perception multicouche). Le rôle de cette partie est de combiner les caractéristiques du code CNN pour classer l'image.

La sortie est une dernière couche comportant un neurone par catégorie. Les valeurs numériques obtenues sont généralement normalisées entre 0 et 1, de somme 1, pour produire une distribution de probabilité sur les catégories.

3.3.3.3 Les étapes principales dans CNN

CNN résume en 4 étapes principales :

- 1) Convolution.
- 2) Sous-échantillonnage (subsampling).
- 3) Activation.
- 4) Connexion complète.

a) Étape 1 : Convolution

Les premières couches qui reçoivent un signal d'entrée sont appelées filtres de convolution. Convolution est un processus dans lequel le réseau tente de marquer le signal d'entrée en se référant à ce qu'il a appris dans le passé.

La convolution a la belle propriété d'être invariante translationnelle. Intuitivement, cela signifie que chaque filtre de convolution représente une caractéristique intéressante, et l'algorithme CNN apprend quelles fonctionnalités comprennent la référence résultante. La puissance du signal de sortie ne dépend pas de l'emplacement des fonctionnalités, mais simplement de la présence des caractéristiques.

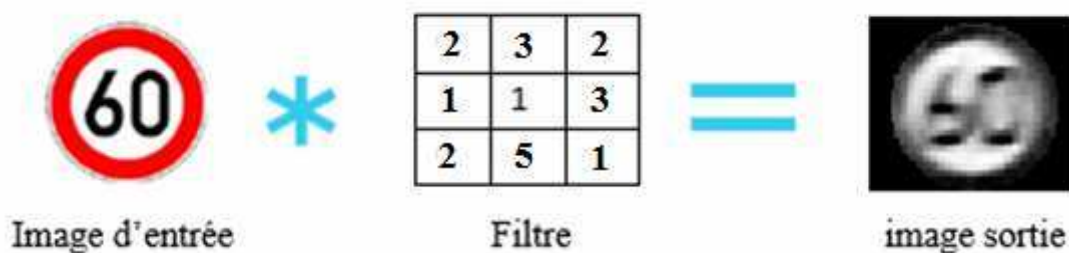


Figure 19 : Convolutionner l'image avec un filtre de cercle

b) Étape 2 : sous-échantillonnage

Les entrées de la couche de convolution peuvent être "lisses" pour réduire la sensibilité des filtres au bruit et aux variations. Ce processus de lissage s'appelle sous-échantillonnage et

peut être réalisé en prenant des moyennes ou en prenant le maximum sur un échantillon du signal. Des exemples de méthodes de sous-échantillonnage (pour les signaux d'image) incluent la réduction de la taille de l'image ou la réduction du contraste de couleur sur les canaux rouge, vert, bleu (RGB).

c) **Étape 3 : activation**

La couche d'activation contrôle la façon dont le signal passe d'une couche à l'autre, imitant la façon dont les neurones sont tirés dans notre cerveau. Les signaux de sortie qui sont fortement associés aux références passées activeront plus de neurones, permettant aux signaux de se propager plus efficacement pour l'identification.

CNN est compatible avec une grande variété de fonctions d'activation complexes pour modéliser la propagation du signal, la fonction la plus courante étant l'unité linéaire rectifiée (ReLU), qui est favorisée pour sa vitesse d'entraînement plus rapide.

d) **Étape 4 : Entièrement connecté**

Les dernières couches du réseau sont entièrement connectées, ce qui signifie que les neurones des couches précédentes sont connectés à chaque neurone dans les couches ultérieures. Cela imite le raisonnement de haut niveau où toutes les voies possibles de l'entrée à la sortie sont prises en considération.

e) **(Pendant la formation) Étape 5: Perte**

Lors de la formation du réseau neuronal, il existe une couche supplémentaire appelée couche de perte. Cette couche fournit des commentaires sur le réseau de neurones sur la façon dont il identifiait les entrées correctement et, dans le cas contraire, dans la mesure du possible. Cela aide à guider le réseau neuronal pour renforcer les bons concepts au fur et à mesure qu'il s'entraîne. C'est toujours la dernière couche pendant la formation.

Le réseau neuronal convolutif est une technique d'apprentissage approfondi populaire pour les tâches de reconnaissance visuelle actuelles. Comme toutes les techniques d'apprentissage profond, CNN est très dépendante de la taille et de la qualité des données de formation. Compte tenu d'un ensemble de données bien préparé, les CNN sont capables de surpasser les humains dans les tâches de reconnaissance visuelle. Cependant, ils ne sont toujours pas robustes aux artefacts visuels tels que l'éblouissement et le bruit, que les humains peuvent supporter. La théorie de la CNN est en cours de développement et les chercheurs travaillent à la doter de propriétés tels que l'attention active et la mémoire en ligne, ce qui permet aux CNN d'évaluer de nouveaux éléments qui sont très différents de ce qu'ils ont été formés. Cela émule mieux le système visuel des mammifères, évoluant vers un système de reconnaissance visuelle artificiel plus intelligent.

4. Diagramme de l'application générale de notre application

Ce diagramme général (figure 22) décrit les différentes solutions éprouvées durant nos tests afin de déterminer quelle est celle qui permet d'obtenir le meilleur taux de reconnaissance pour aider au mieux les conducteurs lors de la prise de décision. Ainsi à partir d'une base de données contenant des images préchargées des différents panneaux routiers, nous avons testé et comparé les résultats obtenus par ces différents algorithmes de classification :

- MLP sans utiliser de prétraitement,
- MLP avec le prétraitement LBP,
- MLP avec le prétraitement ACP,
- MLP avec le prétraitement HOG,
- CNN sans utiliser de prétraitement.

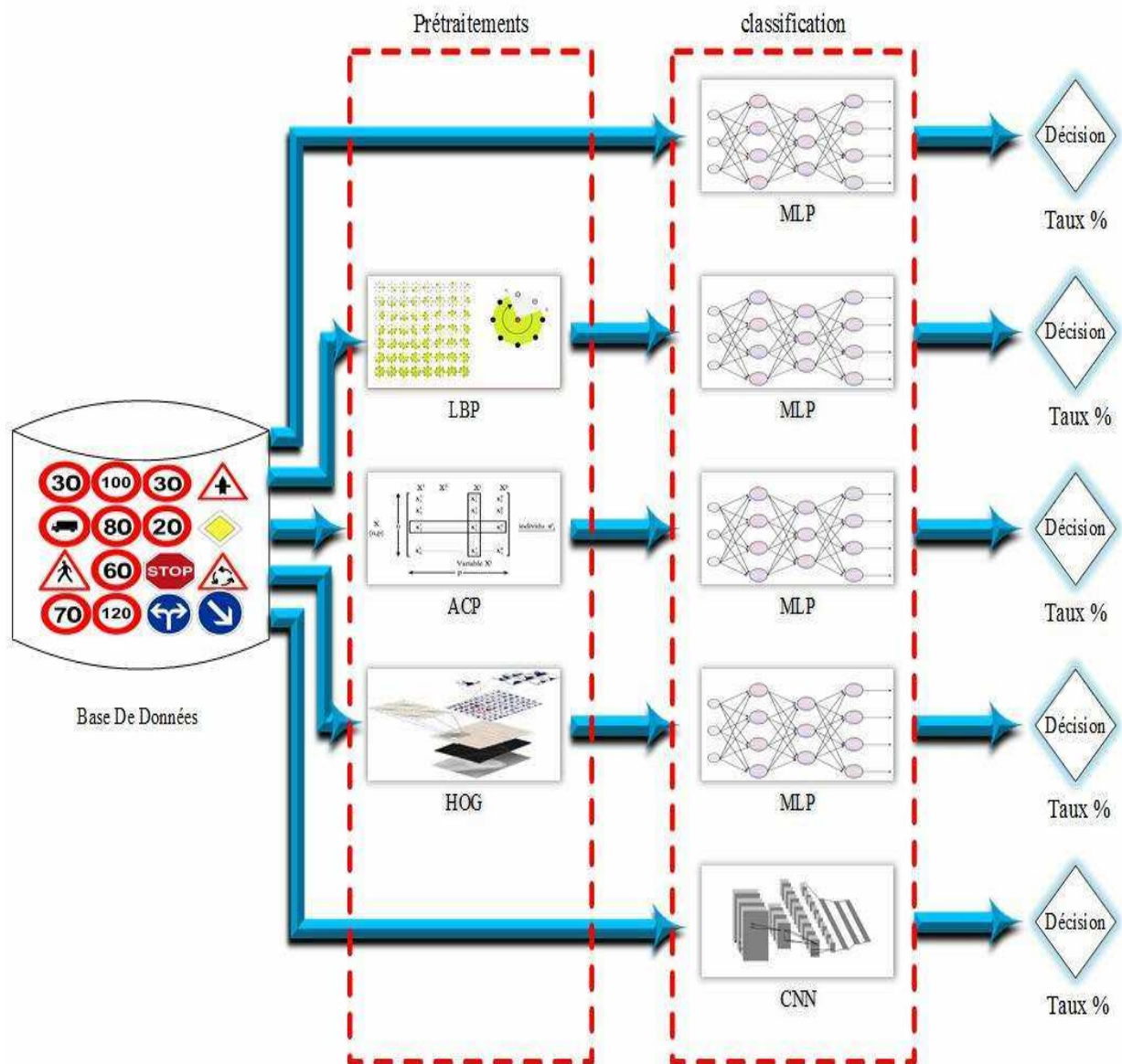


Figure 20 : Diagramme général de notre application

Conclusion

Dans ce chapitre nous avons présenté une étude générale sur la reconnaissance des panneaux routiers. Nous avons en premier lieu discuté les principaux aspects liés à la reconnaissance des panneaux routiers d'une manière générale. Ensuite, nous avons abordé chaque étape du modèle (Acquisition et prétraitement, extraction des caractéristiques, classification, décision). En précisant les principales méthodes de reconnaissance

1. Introduction

Après avoir pris connaissance dans les chapitres précédents des différents descripteurs et classifieur utilisé pour les TSR, nous présentons dans ce chapitre les outils utilisés pour le développement de notre système, et sa mise en pratique, ainsi que les différents paramètres utilisés pour l'extraction des caractéristiques, et la stratégie utilisée pour la classification. Nous validons notre travail par la description de différentes expérimentations réalisées et les résultats obtenus.

2. Ressources utilisées

2.1 Les ressources physiques exploitées :

- Processeur Intel(R) Core(TM) i5-2350M CPU @ 2.30GHZ / mémoire vive d'une capacité de 8Go.
- Processeur Intel(R) Core(TM) i7-2350M CPU @ 2.30GHZ / mémoire vive d'une capacité de 8Go.

2.2 Les ressources logicielles, nous avons utilisé :

- Système d'exploitation : Windows10.
- Langage de programmation : Matlab 2016

2.3 Pour quoi Matlab :

MATLAB (« *matrix laboratory* ») 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.

Le code d'exemple Matlab fournit des fonctions pour itérer sur les jeux de données (formation et test) pour lire les images et les annotations correspondantes. Les emplacements où nous pouvons joindre notre méthode d'entraînement ou de classification sont marqués dans le code par des appels de fonctions fausses.

3. La base de données :

Les panneaux de signalisation sont caractérisés par une grande variabilité dans leur apparence visuelle dans des environnements du monde réel. Par exemple, les changements d'éclairage, des conditions météorologiques variables et occlusions partielles impactent sur la perception de la signalisation routière (Figure 21). Dans la pratique, un grand nombre de différentes classes de panneaux doit être reconnu avec une très grande précision. Les panneaux de signalisation ont été conçus pour être facilement lisibles pour les humains, qui exécutent très bien cette tâche. Cependant, pour les systèmes informatiques, la classification des panneaux de signalisation pose un problème de reconnaissance des formes difficiles.

Le benchmark allemand de la reconnaissance des panneaux routiers GTSRB (German TrafficSign Recognition Benchmark) [w3] présente une base de données possède les propriétés suivantes :

3.1 Aperçu

- Problème de classement à une seule image et multiclasse
- 43 classes
- Plus de 50 000 images au total
- Une grande base de données personnalisée
- Des données de vérité fondamentale fiables en raison de l'annotation semi-automatique
- Les instances de signalisation physique sont uniques dans l'ensemble de données (c'est-à-dire que chaque signal de circulation du monde réel ne se produit qu'une seule fois)



Figure 21 : Les panneaux routiers GTSRB

3.2 Format d'image

- Les images contiennent une enseigne de circulation
- Les images contiennent une bordure de 10% autour de la carte de signalisation réelle (au moins 5 pixels) pour permettre des approches basées sur les bordures
- Les images sont stockées au format PPM
- Les tailles d'image varient entre 15x15 et 250x250 pixels.
- Les images ne sont pas nécessairement carrées.
- Le trafic réel n'est pas forcément centré dans l'image. Ceci est vrai pour les images proches de la bordure de l'image dans l'image complète de la caméra.
- La boîte de délimitation de la circulation fait partie des annotations.

4. L'extraction des descripteurs

Pour l'extraction de descripteurs, nous avons utilisé trois méthodes différentes, LBP, ACP et HOG

5. Classification des descripteurs

Pour la classification des descripteurs, nous avons utilisé deux méthodes différentes, réseau de neurones multicouche (MLP), réseau de neurones convoluionel (CNN)

6. Résultats obtenus

6.1. Approche 1 : classification avec MLP

Comme mentionné avant, la reconnaissance des panneaux routiers passe par deux Phases, l'extraction des caractéristiques et leur classification.

Lors de cette phase d'essai, nous avons effectué 20 tests par algorithme, il y en 4 différents, soit 80 tests au total, et ce uniquement pour la classe 43. Nous avons commencé par tester l'algorithme MLP seul, puis ce même algorithme associé à différents prétraitements : LBP, ACP et HOG. Ces tests ont pour but de nous aider à définir quel est l'algorithme qui nous permettra d'obtenir le meilleur taux de reconnaissance. Chacun des algorithmes a été testé à 25%, 40%, 50% et 70% de sa capacité maximale. Comme vous pouvez le constater, chacune des colonnes nous montre le taux de reconnaissance obtenu en fonction du paramètre « couches cachées » utilisé.

Nous pouvons lire par exemple, que le taux de reconnaissance s'élève à 91.23% en utilisant l'algorithme MLP à 25%, et en utilisant le paramètre « couches cachées = 40 », ou qu'il est de 84,63% en utilisant l'algorithme MLP associé au prétraitement HOG à 70% avec le paramètre « couches cachées = 800 ».

6.1.1. MLP avec une image brute

Ainsi nous pouvons constater que le meilleur taux de reconnaissance obtenu pour l'algorithme MLP est de 92,50%. Ce taux est le plus élevé et il a été obtenu lors du test à 25% grâce au paramètre « couches cachées = 100 ».

		25%		40%		50%		70%	
		9349 Images tests	28048 Images train	14958 Images tests	22439 Images train	18698 Images tests	18698 Images train	26177 Images tests	11220 Images train
Les couches cachet	40	91.23		91.09		90.58		87.48	
	100	92.50		91.69		90.48		88.01	
	200	92.31		92.36		90.72		87.68	
	400	92.45		92.05		88.58		86.69	
	800	91.67		91.57		89.86		85.82	

Tableau 3 : Taux de reconnaissance (%) MLP avec une image brute

6.1.2. MLP avec un prétraitement LBP

Pour ce qui est de l’algorithme MLP_LBP, le meilleur taux de reconnaissance est de 96,84%. Ce taux est le plus élevé et il a été obtenu lors du test à 25% grâce au paramètre « couches cachées = 100 ».

		25%		40%		50%		70%	
		9349 Images tests	28048 Images train	14958 Images tests	22439 Images train	18698 Images tests	18698 Images train	26177 Images tests	11220 Images train
Les couches Cachet	40	95.98		96.33		95.72		93.65	
	100	96.84		96.69		96.43		94.69	
	200	96.72		96.89		96.52		94.7	
	400	96.53		96.64		96.22		94.44	
	800	95.89		95.84		95.52		93.55	

Tableau 4 : Taux de reconnaissance (%) MLP avec un prétraitement LBP

6.1.3. MLP avec un prétraitement ACP

Concernant l’algorithme MLP_ACP, le meilleur taux de reconnaissance obtenu est de 48,11%. Ce taux est le plus élevé et il a été obtenu lors du test à 25% grâce au paramètre « couches cachées = 400 ».

		25%		40%		50%		70%	
		9349 Images tests	28048 Images train	14958 Images tests	22439 Images train	18698 Images tests	18698 Images train	26177 Images tests	11220 Images train
Les couches cachet	40	38.97		34.70		33.98		37.88	
	1000	40.50		42.44		34.14		34.23	
	200	42.04		43.67		34.47		34.33	
	400	48.11		44.58		37.80		33.58	
	800	46.13		44.61		40.81		32.26	

Tableau 5 : Taux de reconnaissance (%) MLP avec prétraitement ACP

6.1.4. MLP avec un prétraitement HOG

Aussi pour l’algorithme MLP_HOG, le meilleur taux de reconnaissance obtenu est de 91,01%. Ce taux est le plus élevé et il a été obtenu lors du test à 25% grâce au paramètre « couches cachées = 400 ».

		25%		40%		50%		70%	
		9349 Images tests	28048 Images train	14958 Images tests	22439 Images train	18698 Images tests	18698 Images train	26177 Images test	11220 Images train
Les couches cachet	40	83.66		83.35		82.41		79.73	
	100	87.43		85.90		83.34		80	
	200	88.55		86.20		86		81.32	
	400	91.01		87.33		88.8		85.26	
	800	87.95		86.42		88.13		84.63	

Tableau 6 : Taux de reconnaissance (%) MLP avec un prétraitement HOG

Après cette batterie de tests, le meilleur résultat obtenu l’a été pour l’algorithme MLP_LBP (avec le prétraitement LBP), le taux de reconnaissance a atteint le score de 96,84% malgré un nombre de couches cachées peu élevé (ici le nombre de couches cachées était de 100). Ce prétraitement LBP a permis d’accroître les performances de l’algorithme MLP d’environ 4 à 8%.

Nous pouvons noter que durant les différents tests, et ce pour les 4 algorithmes testés, les meilleurs taux de reconnaissance obtenus l’ont toujours été lors des tests à 25% de leur capacité respective. Et que plus cette capacité utilisée augmente, plus le taux de reconnaissance diminue.

Nous avons également constaté lors des tests que plus nous augmentons le nombre de couches cachées utilisées pour un même algorithme, plus le test met du temps à s’effectuer du fait du nombre élevé d’informations qu’il a à traiter.

Il est intéressant de voir qu’à partir d’un certain seuil, l’augmentation du nombre de couches cachées devient contre-productive et nous donne des résultats moins performants. Par exemple pour l’algorithme MLP à 25%, le passage de 400 à 800 couches cachées fait diminuer le score obtenu. Idem à 40%, 50% et 75%, le passage de 200 à 400 fait baisser les résultats. Ce phénomène est visible sur tous les algorithmes testés.

Pour ce qui est des algorithmes MLP et MLP_LBP, les meilleurs résultats ont été obtenus en utilisant le nombre de couches cachées = 100.

Alors que pour les deux autres algorithmes MLP_ACP et MLP_HOG, les meilleurs résultats ont été obtenus en utilisant le nombre de couches cachées = 400.

Et enfin, l’utilisation des prétraitements HOG et surtout ACP permettent d’accélérer les tests, mais ce gain en temps se fait en dépit du taux de reconnaissance qui chute drastiquement

pour MLP_ACP et diminue pour MLP_HOG, si l'on compare ces résultats à ce obtenu avec MLP seul.

En conclusion, nous pouvons dire que chacun de ces algorithmes à ses qualités, et que si la vitesse d'exécution est privilégiée en dépit du taux de reconnaissance, il est préférable de choisir l'algorithme MLP associé au prétraitement ACP ou HOG, alors que si nous privilégions les meilleurs taux de reconnaissance à la vitesse, mieux vaut utiliser MLP associé au prétraitement LBP voir MLP seul. Enfin l'augmentation du nombre de couches cachées permet d'améliorer les résultats dans une certaine mesure tout en veillant à ne pas dépasser un seuil propre à chaque algorithme et au pourcentage utilisé, sinon les résultats seront contre-productifs avec des délais de traitement assez long et des taux de reconnaissance plus faibles.

6.2. Approche 2 : classification avec CNN

Lors de cette phase d'essai, nous avons effectué 5 tests avec l'algorithme CNN, en modifiant à chaque fois le nombre d'itérations. Ainsi nous l'avons testé avec 1 itération, puis 50 itérations, ensuite 100, 200 et enfin 300 itérations. Ces différents tests nous affichent les résultats obtenus ainsi que le temps d'exécution mis par chacun dans le tableau intitulé (table)

Nombre d'itération	Résultats	Temps d'exécution
1	05.7%	20 minutes
50	40.03	40 minutes
100	67.55	24 heures
200	81.5	36 heures et 30 minutes
300	93.5	48 heures

Tableau 7 : Taux de reconnaissance (%) et temps d'exécution de l'algorithme CNN en fonction du nombre d'itérations

Nous pouvons lire par exemple, que le taux de reconnaissance s'élève à 5.7% en utilisant l'algorithme CNN avec une seule itération, et qu'il a mis 20 minutes pour s'exécuter. Et que pour obtenir le score de 67,55%, il a fallu utiliser CNN avec 100 itérations pendant 24 heures.

À la suite de ces tests, nous pouvons voir que le meilleur taux de reconnaissance (93,5%) est obtenu en utilisant l'algorithme CNN avec 300 itérations, et qu'il a duré 48 heures. Nous pouvons constater que plus le nombre d'itérations utilisées augmente, plus le temps d'exécution s'allonge et plus le résultat obtenu est élevé.

6.3. Comparaison entre MLP et CNN

Après un aperçu des résultats des différents tests effectués, nous pouvons affirmer que l'algorithme MLP est moins performant que l'algorithme CNN, qui est lui-même moins efficace que l'algorithme MLP associé au prétraitement LBP. Car le premier obtient comme










meilleur taux de reconnaissance **92,50%** lors du test à 25% grâce « 100 couches cachées ». Alors qu’avec CNN, nous obtenons **93,5%**. Enfin avec MLP avec LBP nous obtenons **96,84%** lors du test à 25% grâce au paramètre «100 couches cachées ».

7. Validation

Pour valider notre application nous avons calculé la matrice de confusion pour chacun des algorithmes la matrice de confusion, dans la terminologie de l’apprentissage supervisé, est un outil servant à mesurer la qualité d’un système de classification. Chaque ligne de la matrice représente le nombre d’occurrences d’une classe estimée, tandis que chaque colonne représente le nombre d’occurrences d’une classe réelle (ou de référence).

Dans notre système, les lignes 1 jusqu’à 43 (successivement) représentent le nombre d’occurrences des classes estimé. (Tableau 8), et les colonnes 1 jusqu’à 43 (successivement) représentent le nombre d’occurrences des classes réelles.

Un des intérêts de la matrice de confusion est qu’elle montre rapidement si le système parvient à classer correctement ou pas.

Classe	Nom de classe	Le panneau
01	Accès interdit aux véhicules affectés au transport de marchandises.	
02	Annonce d'un endroit fréquenté par les enfants.	
03	Annonce d'un virage à droite.	
04	Annonce d'un virage à gauche.	
05	Annonce d'une chaussée glissante.	
06	Annonce d'une chaussée rétrécie.	
07	Annonce de Danger.	
08	Annonce de cyclistes venant de droite ou de gauche.	
09	Annonce de feux tricolores.	

10	Annonce de passage d'animaux sauvages.	
11	Annonce la proximité d'un cassis.	
12	Circulation interdite à tout véhicule dans les deux sens.	
13	Contournement obligatoire par la droite.	
14	Contournement obligatoire par la gauche.	
15	Direction obligatoire à la prochaine intersection toute droite.	
16	Direction obligatoire à la prochaine intersection à droite.	
17	Direction obligatoire à la prochaine intersection à gauche.	
18	Fin d'interdiction de dépasser notifiée par le panneau.	
19	Fin d'interdiction de dépasser notifiée par le panneau.	
20	Fin de limitation de vitesse 80.	
21	Fin de toutes les interdictions précédemment signalées.	

22	Interdiction aux véhicules de transports de marchandises de doubler.	
23	Interdiction de dépasser tous les véhicules à moteur autres que ceux à deux roues sans side-car.	
24	Panneau de danger neige.	
25	Sens interdit à tout véhicule.	
26	Signification Annonce d'une intersection avec une route.	
27	Travaux.	
28	Céder le passage.	
29	Signification Annonce ou rappel du début d'une route à caractère prioritaire.	
30	Panneau de signalisation indiquant le sens de la rotation dans un système de circulation à droite.	
31	Limitation vitesse 100.	
32	Limitation vitesse 120.	












33	Limitation vitesse 20.	
34	Limitation vitesse 30.	
35	Limitation vitesse 50.	
36	Limitation vitesse 60..	
37	Limitation vitesse 70.	
38	Limitation vitesse 80.	
39	Directions obligatoires à la prochaine intersection tout droit ou à droite.	
40	Directions obligatoires à la prochaine intersection tout droit ou à gauche.	
41	Panneaux de danger piéton.	
42	Stop.	
43	Annonce d'une succession de virages dont le premier est à gauche.	

Tableau 8 : les panneaux routiers de la base de données GTSRB

La figure 22 montre la matrice de confusion MLP_LBP :

$$\text{Le taux de reconnaissance} = \frac{\text{la somme de diagonale}}{\text{nombre totale image}} = \frac{9054}{9349} = 0.9684 = 96.84\%$$

8. Conclusion

Ce dernier chapitre qui explique les différents processus et paramètres utilisés lors de nos différents tests nous a permis de classifier les algorithmes de classification grâce aux taux de reconnaissance obtenus. Ainsi, l'algorithme de classification MLP a été expérimenté seul, puis avec différents prétraitements. Tout comme l'algorithme CNN. Nous avons pu voir leurs niveaux de reconnaissance, et le temps de traitement mis par chacune de ces configurations. Il en ressort que le meilleur taux de reconnaissance a été par l'association du prétraitement LBP avec l'algorithme MLP, et ce avec des paramètres peu gourmands en ressources.

Conclusion générale

Dans notre projet, nous nous sommes intéressés à l'utilisation des systèmes de la reconnaissance des formes. Nous avons donné quelques définitions sur RDF d'une façon générale et l'évaluation de la recherche sur la Reconnaissance des panneaux routiers.

Pour ce faire, nous avons testé et comparé différents algorithmes de classification (MLP et CNN) en leur associant ou non différents prétraitements (LBP, ACP, HOG).

Les motivations qui conduisent à faire effectuer des tâches de reconnaissance par des machines sont nombreuses. L'imitation des facultés de reconnaissance de l'homme sera toujours un moteur pour le progrès dans ce domaine. Toutefois il est important de signaler que ceci doit se faire non pas pour remplacer l'homme par une machine, mais pour le soulager des tâches ennuyeuses voir dangereuses.

Le but ce travail était de proposer un système de reconnaissance des formes qui sera utilisé pour reconnaître les panneaux routiers et ainsi faciliter l'aide à la conduite aux automobilistes. Ainsi en le soulageant de cette tâche on pourra éviter des dommages, accidents et améliorer la sécurité des autres usagers de la route (piétons).

Suite à ce travail, de nombreuses perspectives sont envisageables, la plus importante étant sans doute la capacité que devra avoir notre système de reconnaissance des formes à fonctionner dans n'importe quelles conditions, notamment météorologiques (brouillard, pluie, obscurité, forte luminosité . . .). Il est également possible de le perfectionner en optimisant les algorithmes, surtout pour réduire le temps d'exécution et ainsi permettre une application en temps réels à bord de véhicules.

Références bibliographiques

- [1] Likforman-Sulem Laurence, Barney-Smith Élisabeth, IMAGERIE - Reconnaissance des formes - Théorie et pratique sous Matlab - Cours et exercices corrigés ,2013
- [2] L. Miclet, Méthode structurelle pour la reconnaissance des formes. EYROLLES, collection technique et scientifique de télécommunication, 1984.
- [3] Committee of CNNs, IDSIA , Multi-column deep neural network for traffic sign classification, Multi-column deep neural network for traffic sign classification, D. Cireşan, U. Meier, J. Masci, J. Schmidhuber, August 2012, Neural Networks (32), pp. 333-338
- [4] Color-blob-based COSFIRE filters for object recogn, COSFIRE, Color-blob-based COSFIRE filters for Object Recognition, Color-blob-based COSFIRE filters for Object Recognition, Baris Gecer, George Azzopardi, Nicolai Petkov, 2017, Image and Vision Computing (57), pp. 165-174
- [5] Human Performance, INI-RTCV , Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition, Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition, J. Stallkamp, M. Schlipsing, J. Salmen, C. Igel, August 2012, Neural Networks (32), pp. 323-332
- [6] Multi-Scale CNNs, sermanet , Traffic sign recognition with multi-scale Convolutional Networks, Traffic sign recognition with multi-scale Convolutional Networks, P. Sermanet, Y. LeCun, August 2011, International Joint Conference on Neural Networks (IJCNN) 2011
- [7] Random Forests, CAOR , Traffic sign classification using K-d trees and Random Forests , Traffic sign classification using K-d trees and Random Forests , F. Zaklouta, B. Stanculescu, O. Hamdoun, August 2011, International Joint Conference on Neural Networks (IJCNN) 2011
- [8] LDA on HOG 2, INI-RTCV, Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition, Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition, J. Stallkamp, M. Schlipsing, J. Salmen, C. Igel, August 2012, Neural Networks (32), pp. 323-332
- [9] G. Vamvakas, B. Gatos, S. J. Perantonis, "Handwritten character recognition through two-stage foreground sub-sampling", Pattern Recognition, Vol. 43
- [10] T. Ojala, Pietikainen, M., & Harwood, D., "A comparative study of texture measures with classification based on feature distributions.," In Pattern Recognition, vol. 29(1), pp. 51–59, 1996.
- [11] Heikkilä M, Pietikäinen M, Schmid C. Description of interest regions with local binary patterns. Pattern Recognit. 2009.

- [12] Dalal et B. Triggs : Histograms of Oriented Gradients for Human Detection. In Proceedings of the International Conference on Computer Vision and Pattern Recognition, 2005.
- [13] G. Lowe: Distinctive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision, 2004.
- [14] Dong Xiao Ni, “Application of Neural Networks to Character Recognition”, Proceedings of Students/Faculty Research Day, CSIS, Pace University, May 4th, 2007.
- [15] D. Montana, L. Davis, “Training feedforward neural networks using genetic algorithms”, in: Proceedings of Eleventh International Joint Conference on Artificial Intelligence, N.S. Sridharan (Ed.), Morgan Kaufman Publishers, 1989.
- [16] D.F.A.Santiago, R. Pederiva, ”Comparison of optimization techniques of neural networks training for faults diagnostic of rotating machinery”. Proceeding of the 1st South-American Congress on Computational Mechanics, MECOM, pp: 1912-1921. 2002
- [17] Garcia-Garrido, M. A., Sotelo, M. A. and Martin-Gorostiza, E., 2006. Fast Traffic Sign Detection and Recognition under Changing Lighting Conditions. In: IEEE International Conference on Intelligent Transportation Systems, Toronto, Ont., Canada, pp. 811–816.
- [18] Gavrilu, D. M. Traffic Sign Recognition Revisited. In: DAGM Symposium für Mustererkennung, Bonn, Germany, pp. 86–93. 1999

Webography

Liste des Références webographies

- [w1] <http://voiture.kidioui.fr/lexique-automobile/reconnaissance-des-panneaux>
- [w2] <http://blog.octo.com/classification-dimages-les-reseaux-de-neurones-convolutifs-en-toute-simplicité>
- [w3] <http://benchmark.ini.rub.de?section=gtsrb&subsection=dataset#Downloads>