

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE
SCIENTIFIQUE



Université Mouloud MAMMERRI - TIZI OUZOU

Faculté de Génie Electrique et Informatique

Département d'Informatique

Mémoire de fin d'études

En vue de l'obtention du diplôme de master en Informatique

Option : RMSE

Thème :

Classification d'images agricoles avec le Deep Learning

Realisé par :

Dyhia BOUADI

Thanina IMARAZENE

Encadré par : Mme Rachida AOUDJIT

Devant le jury composé de :

- **Mme Malika BELKADI**

- **Mr Mohammed DAOUI**

Année : 2018/2019

Tables des matières :

Introduction générale.....	12
Problématique et Objectifs.....	13
Plan du mémoire.....	14

Chapitre 1 : Agriculture de précision

1. Aspects et Concepts de l’Agriculture de Précision (AP).....	16
1.1 L'agriculture de précision	16
1.2 Le principe de l’agriculture de précision.....	17
1.3 Adoption de l’agriculture de précision :.....	17
1.4 La télédétection /Imagerie et l’agriculture de précision	18
1.5 Avantages et impacts de l’agriculture de précision.....	18
2. Les drones et leurs relations avec l’agriculture de précision (AP).	19
2.1 Les drones.....	19
2.2 Composants d’un drone	20
2.3 Les capacités du drone.....	21
2.4 Typologie de drones	22
2.5 Les fonctions du drone	23
2.6 Les drones au service de l'agriculture.....	24
2.7 La cartographie NDVI, un outil d’une extrême efficacité.....	25
2.8 Méthodes d’optimisation et réduction	26
Conclusion.....	27

Chapitre 2 :Deep Learning ,Machine Learning et Reseaux de neurones et classification d'images

Introduction.....	29
1. Deep Learning, Machine Learning et les réseaux d neurones.	30
1.1 Principe de l’Apprentissage Automatique (Machine Learning).....	30
1.2 Qu'est-ce que l'apprentissage ?	31
1.3 Principe de l’Apprentissage Profond (Deep Learning)	32
1.4 Historique de Deep Learning :.....	33
1.5 Pourquoi le Deep Learning ?.....	34
1.6 Fonctionnement du Deep Learning	35
1.7 Domaine d’applications du <i>Deep Learning</i> [5] :.....	35

1.8	Les Réseaux de Neurones.....	36
1.8.1	Modélisation d'un neurone : une inspiration biologique :	36
1.8.2	Principe des réseaux de neurones.....	38
1.8.3	Typologie des réseaux neuronaux :.....	38
1.8.3.1	Neurone Formel	39
1.8.3.2	Neurones multicouches.....	39
1.8.3.3	Neurones récurrents	40
1.8.3.4	Réseaux Neurones Convolutionnels (CNN).....	40
2.	Réseaux de Neurones Convolutifs (CNNs).....	41
2.1.	Principe d'architecture d'un CNN.....	41
2.2.	Qu'est-ce que la convolution ?	41
3.	Les blocs de construction d'un apprentissage en profondeur.....	43
3.1	Couche d'image d'entrée (Image Input Layer)	43
3.2	La couche de convolution (convolutionnet Layer) [23] :.....	44
3.3	La Couche de regroupement (POOL)	45
3.4	Couche de correction [26] :.....	46
3.5	Couche entièrement connectée (FullyconnectedLayersFC).....	50
3.6	Couche de perte (LOSS)	50
4.	Choix des hyperparamètres.....	50
4.1	Nombre de filtres	51
4.2	Forme de filtres	51
4.3	Forme de pooling.....	51
5.	Différence entre réseaux de neurones convolutifs comparés et perceptron multicouche.....	53
5.1.	Avantages de CNNs.....	53
6.	Approches & Méthodes de Classification d'Images.....	53
6.1	Définition de la classification	53
6.2	Les motivations de la Classification des images	54
6.3	Différentes méthodes de la classification et l'apprentissage	55
6.3.1	Méthodes supervisées.....	55
6.3.2	Méthodes non supervisées.....	55
6.4	Indicateurs de performance en classification.....	56
6.4.1.	Matrice de confusion :	58
7.	Classification des images et l'apprentissage machine.....	59
8.	Classification des images et les réseaux de neurones	60

9.	Deep Learning et agriculture.....	62
	Conclusion :	61
Chapitre 3 : Implémentation d'une application de classification		
	Introduction :.....	64
1.	Configuration logiciel utilisé dans l'implémentation :.....	65
2.	Configuration matérielle utilisé dans l'implémentation :.....	66
3.	Qu'est-ce qu'un réseau pré-entraîné ?	66
4.	Les réseaux de neurones pré-entraînés :	68
4.1	Alexnet :	68
4.2	Resnet-50 :	68
5.	Architecture de notre réseau :.....	68
5.1	Architecture du modèle 01 (AlexNet) :	69
5.2	Architecture du modèle 02 (Resnet-50) :	70
6.	Présentation de l'application :	73
7.	Les bases d'images :	73
7.1	Dataset1:	74
7.2	Dataset 2 :	74
8.	Pré traitement des données.....	76
9.	Entraînement du modèle.....	77
9.1.	Sur-apprentissage et sous-apprentissage	78
9.2.	Visualisation des filtres Convnet	79
9.3.	Axe d'amélioration	82
9.4.	C'est quoi Data augmentation ?.....	85
10.	Matrice de confusion.....	86
10.1	Comprendre les terminologies de la Confusion Matrix.....	85
10.2	Réaliser une prédiction	88
11.	Courbe ROC :.....	91
12.	Les principales lignes du code et leurs fonctionnalité.....	96
	Conclusion.....	96
	Conclusion générale	97
	Perspectives.....	98
	Références Bibliographique	98

Liste des figures

Chapitre 1 :

Figure 1 : utilisation de drone pour l'agriculture	17
Figure 2:Image représentative d'un drone	21
Figure 3:Images technique réalisées pas un drone	25
Figure 4: Photo d'un drone pulvérisateur	27

Chapitre 2 :

Figure 5: Les différents algorithmes de machine learning	30
Figure 6:La relation entre l'intelligence artificielle, ML et deeplearning	32
Figure 7: La différence de performance entre le Deeplearning et la plupart des algorithmes de ML en fonction de la quantité de données	34
Figure 8;Le procédé du ML classique comparé à celui du Deep Learning.....	36
Figure 9:Schémas d'un neurone biologique	37
Figure 10: schémas d'un neurone formel.....	37
Figure 11:représentation d'un perceptron multi-couche.....	39
Figure 12: principe général d'un réseau e neurones récurrents	40
Figure 13:Architecture d'un réseau de neurones convolutif.....	41
Figure 14:Schéma du parcours de la fenêtre de filtre sur l'image	42
Figure 15:Exemple d'une convolution	43
Figure 16:Une couche du CNN en 3 dimensions. (Vert = volume d'entrée, bleu = volume du champ récepteur, gris = couche de CNN, cercles = neurones artificiels indépendants)	45
Figure 17:Pooling avec un filtre 2x2 et un pas de 2.....	46
Figure 18:Nombre de modèles de machine Learning pour des usages agricoles d'après etude couvrant une période de 2004 a nos jours- Source	60

Chapitre 3 :

Figure 19:Environnement de Matlab.....	65
Figure 20:Le graphique précision de la classification en fonction du temps de prédiction avec un GPU	67
Figure 21: Architecture AlexNet.....	70
Figure 22 : bloc résiduel.....	71
Figure 23:schéma d'un réseau résiduel et un réseau simple	71
Figure 24:Architecteur de ResNet 50.....	72
Figure 25: Echantillon d'images classe A	74
Figure 26:Echantillon d'image classe B	74
Figure 27: Echantillon d'images de classe maïs	75
Figure 28 : Echantillon d'images classe betterave.....	75
Figure 29: graphe de suivit de l'entraiment de AlexNet.....	78

Figure 30::exemples de filtres à plusieurs niveaux du CNN : au début, au milieu (après quelques convolutions) et à la fin.....	80
Figure 31: 96 filtres de la première couche de convolution de AlexNet.....	80
Figure 32:64 filtres de la première couche de convolution de ResNet50	80
Figure 33:Image d'entrée (227*227) et application d'un filtre AlexNet	81
Figure 34:résultat d'application du filtre et un étirement de contraste pour mieux visualiser .	83
Figure 35:résultat d'application de la couche ReLu	84
Figure 36: Exemple d'augmentation avec rotation et ajout de contraste, luminosité a l'image original	84
Figure 37:graphe d'apprentissage de resnet50	84
Figure 38: matrice de confusion de AlexNet	86
Figure 39:matrice de confusion du modèle ResNet 50	87
Figure 40: courbe ROC de ResNet 50 apres amélioration	92
Figure 41:Courbe ROC de AlexNet avant amélioration	92

Lister des tableaux :

Tableau 1 : différences entre apprentissage supervisé et non supervisé	56
Tableau 2 :resutlat avant augmentation du jeu de données	88
Tableau 3: résultat apres augmentation du jeu de données	88

Liste des abréviations :

AP : Agriculture de Précision

RGB (RVB) : Rouge vert bleu

IA : Artificial Intelligence

ML : Machine Learning

PNG : Portable Network Graphics

FPV :First Person View

DL :deep Learning

MLP : multi layer perceptron

CNN :Convolutional Neural Networks

ANN :Artificial Neural Network

GPS :GlobalPositioningSysteme

RNN : Recurrent Neural Network

SVM : Support Vector Machine

ROC : Receiver Operating Characteristic

BDD : Base de donnée

RN :Réseaux de neurones

ANN :Artificial Neural Network

ReLU : Rectifiedlinear unit

FC :couches entièrement connectées

TP : True Positives

FP : False Positive

TN :TrueNegatives

FN :FalseNegative

Remerciements :

Tout travail de recherche n'est jamais totalement l'œuvre d'une seule personne, à cet effet, je tiens à exprimer ma sincère reconnaissance et mes remerciements à tous ceux qui ont contribué de près ou de loin à l'élaboration de ce travail. Tout d'abord à Madame AOUIJIT , directrice de ce mémoire, pour l'aide et le temps qu'elle a bien voulu me consacrer et que je ne remercierai jamais assez pour son soutien et sa patience. Qu'elle trouve en ces lignes l'expression de ma gratitude.

Ensuite à Mesdames et Messieurs les membres du jury qui ont eu l'amabilité d'accepter d'évaluer ce travail. Qu'ils trouvent ici l'expression de ma reconnaissance

Enfin, j'adresse mes plus sincères remerciements à tous mes proches et amis qui m'ont toujours soutenue et encouragée au cours de la réalisation de ce mémoire.

Dédicace:

Je dédie ce mémoire à mes chers

Parents pour leur patience

*A ma mère, la lumière de notre vie, qui m'a soutenu et encouragé
durant ces années d'études*

*A mon très cher père, ma Gratitude ne suffit pas à exprimer ce qu'il
mérite pour tous les sacrifices depuis ma naissance, pendant mon
enfance et même à l'âge adulte*

À mes sœurs Sonia, Fette, Sabrina, A mon ferer Karim

*À tous mes ami(e) qui m'ont soutenu dans l'accomplissement de cet
humble travail*

*À tous mes professeurs et à tous ceux qui se sont engagés dans ces
modestes travaux*

À tout ma famille.

Dyhia

Dédicace:

*Je dédie ce mémoire à mes chers
Parents maman et papa qui m'ont encouragé dans chaque pas que
j'ai entrepris durant mes études,
Mes Sœurs Celia et Ghania ainsi que mon petit frère Smail ,
Ainsi que mes amis et ma famille du côté maternelle et paternelle.
Un remerciement spécial à mon cher grand père papa ramdane.*

Thanina

Introduction générale

Introduction générale :

Dans l'optique de voir la population mondiale augmenter grandement dans les prochaines années, l'agriculture va devoir subvenir à des besoins de plus en plus croissants, notamment dans les cultures.

Ainsi l'Homme doit trouver des nouveaux moyens permettant un rendement bien plus élevé. Celui-ci peut être augmenté de plusieurs manières, notamment par une extension du territoire cultivable. Cependant, cette solution ne pourra pas complètement nourrir les générations futures, vu que les surfaces agricoles sont limitées en espace et que l'expansion urbaine fait prendre de plus en plus de place à la ville. L'amélioration du rendement d'une parcelle grâce à la technologie apparaît donc comme un moyen utile pour subvenir à ces futurs besoins. Parmi ces technologies, on retrouve l'agriculture de précision.

L'agriculture de précision est une continuité de la révolution verte, c'est à dire la politique menée par différents pays du monde dans les années 60-90 et qui vise à intensifier les rendements de l'agriculture, notamment en choisissant les variétés de plantes les plus rentables et en utilisant des pesticides et autres intrants (produits apportés à la terre comme l'engrais, les semences, les amendements, les produits phytosanitaires etc...). Elle permet de mieux comprendre et analyser les besoins physiologiques des cultures et de développer des outils d'aide à la décision pour l'utilisateur .

De nos jours, l'agriculture nécessite une compréhension des besoins de chaque parcelle, voire même de chaque plante [7]. Parmi les moyens utilisés par l'agriculture de précision on a la télédétection qui fournit des données très utiles pour identifier et surveiller les cultures, Elle constitue un moyen d'acquisition de données géographique de plus en plus intéressant, qui évolue vers les grandes échelles. Avec les résolutions actuelles, elle est particulièrement bien adaptée à l'analyse de parcelles agricoles, mais aussi à de nombreuses autres applications à petite échelle: prévision de récoltes à l'échelle régionale ou mondiale, analyses statistiques, suivi des évolutions d'un territoire, ...

Les sélectionneurs peuvent l'utiliser pour caractériser les plantes plus facilement et rapidement en réduisant la main d'œuvre sur des tâches souvent fastidieuses (comptages manuels, détection de maladies ...).

Notre travail s'inscrit dans le cadre de la classification d'images en utilisant Le deep learning qui est une technique d'apprentissage permettant à un programme de reconnaître le contenu d'une image, ce système d'apprentissage et de classification est basé sur les réseaux de neurones artificiels (convolutionnels) et cela afin de classer des images agricoles

certaines prises à partir d'un drone et d'autres à partir d'une caméra en plusieurs classes, on va faire recours à différents modèles (réseaux pré-entraînés) avec différentes architectures et par la suite on va appliquer ces modèles sur les bases d'images dataset 1 qui est d'un ensemble d'images agricoles de différentes plantes et qui est divisé en deux classes selon la densité de mauvaises herbes présente dans chaque image (Classe A constituée d'images qui ont une couche dense de mauvaises herbes et Classe B constituée d'images qui ont un taux très réduit de mauvaises herbes) et le dataset 2 contient 12 classes d'images agricoles (Ail, Aubergine, Betterave, Haricot Rouge, Haricot vert, Herbe noire, Lentille, Mais, Moutarde des champs, Poivrier noir, Salade, Tomate).

Problématique et Objectifs

On peut donc définir le but de l'agriculture de précision comme le fait d'augmenter les rendements d'une parcelle agricole tout en réduisant la consommation d'intrants (engrais, phytosanitaires, eau) à apporter grâce à l'aide des nouvelles technologies, autrement dit produire plus avec moins. Dans le domaine de l'AP, la télédétection est assurée par des satellites, des avions, des ballons et des hélicoptères, des petits engins aériens sans pilote (UAS) ou drones et toute une variété de capteurs, tels que des capteurs optiques ou quasi-infrarouge, ou encore des radars. En raison de leur coût peu élevé, les drones pourraient remplacer avantageusement les satellites et les avions. Les agriculteurs peuvent en effet les utiliser pour l'épandage de pesticides, pour suivre les déplacements du cheptel ou assurer le suivi des récoltes et la détection des nuisibles et des maladies.

On peut ainsi se poser la problématique suivante : Drones, satellites... Les technologies peuvent-elles réellement révolutionner l'agriculture ? comment la télédétection sert-elle en agriculture ? comment traiter les images pour qu'elles fournissent de l'information sur les cultures ?

Plan du mémoire

Ce mémoire est divisé en trois principaux chapitres.

Le **chapitre 1** illustre et présente les différents aspects et concepts de l'agriculture de précision et son impact sur la quantité et la qualité du rendement, on a introduit le concept de drone et son impact sur l'AP.

Ensuite **Le chapitre 2** nous présentons l'utilisation du Deep Learning la classification d'image agricole.

Le chapitre 3 est consacré à l'étude des modèles proposé suivi de la conception du modèle étudié. Nous terminons notre rédaction par une conclusion et des perspectives du présent travail.

Chapitre 1
Agriculture de précision

Introduction

Dans ce chapitre nous allons présenter l'agriculture de précision sous ses différents aspects, on va s'intéresser à la télédétection et comprendre son fonctionnement et son utilité dans l'AP. Vues d'en haut, les parcelles ont beaucoup de choses à révéler et depuis l'autorisation de leur utilisation civile, la démocratisation des drones et des technologies d'imagerie aérienne a ouvert la voie à de nombreuses applications métiers dans le secteur agricole. D'une efficacité redoutable, ces aéronefs permettent aujourd'hui le calcul des doses d'intrants, le suivi de la production et la détection de certaines maladies.

Pour les agriculteurs soucieux d'optimiser leur rendement, les drones sont devenus de véritables outils au service d'une agriculture de précision, nous allons voir de plus près ce qu'est le drone et comment ces aéronefs permettent aujourd'hui le calcul des doses d'intrants, le suivi de la production et la détection de certaines maladies. Et leur place dans l'AP. [12]

1 Aspects et Concepts de l'Agriculture de Précision (AP)

1.1 L'agriculture de précision

Est un principe de gestion des parcelles agricoles qui vise l'optimisation des rendements et des investissements, en cherchant à mieux tenir compte des variabilités des milieux et des conditions entre parcelles différentes ainsi qu'à des échelles intra-parcellaires. Dans le contexte de course au progrès des rendements agricoles. Il a notamment influencé le travail du sol, les semis, la fertilisation, l'irrigation, la pulvérisation de pesticides, etc. [9]

Le principe de l'agriculture de précision, c'est d'augmenter les rendements d'une parcelle tout en réduisant la consommation d'énergie et d'intrants. Grâce à l'utilisation des nouvelles technologies, l'idée est ainsi de « *produire plus avec moins* ». L'agriculture de précision utilise des moyens d'observation (satellites, drones, capteurs connectés) associés à des outils d'aide à la décision (accessibles sous forme d'applications Web et Mobile). [2]

Cette nouvelle pratique culturale nécessite des technologies de pointe pour collecter l'information (requiert l'utilisation de nouvelles technologies, telles que l'imagerie satellitaire et l'informatique, s'appuie sur des moyens de localisation dans la parcelle dont le système de positionnement par satellite de type GPS.) et la multiplication des décisions par unité de



Figure 1 : utilisation de drone pour l'agriculture

surface impose souvent l'implication d'outils la parcelle dont le système de positionnement par satellite de type GPS.) et la multiplication des décisions par unité de surface impose souvent l'implication d'outils informatiques complexes pour collecter, enregistrer et exploiter les données pertinentes, de manière à répondre en temps et en quantité aux besoins des cultures dans chaque milieu.

1.2 Le principe de l'agriculture de précision

L'agriculture de précision se base sur :

- La gestion parcellaire et intra-parcellaire ;
- L'optimisation des rendements et des coûts de production ;
- Une meilleure prise en compte du contexte pédoclimatique (nature du sol et météo) ainsi que de la végétation. [11]

1.3 Adoption de l'agriculture de précision :

Le rendement est affecté par de nombreuses variables telles que : le type de sol et la topographie, les précédents, les bios agresseurs présents, l'irrigation, la date et la densité de semis...

Sur bon nombre de ces aspects l'agriculture de précision apporte des solutions concrètes d'amélioration.

Pour utiliser l'agriculture de précision, trois (3) matériels sont indispensables :

- Un système de positionnement GPS.
- Un épandeur à engrais et/ou un pulvérisateur et/ou un semoir doté(s) de la capacité de varier les doses en continu.
- Un boîtier donneur d'ordre pour intégrer les fichiers de modulation et gérer le réglage du matériel en fonction de sa localisation dans la parcelle. [11]

1.4 La télédétection /Imagerie et l'agriculture de précision

Dans le domaine de l'AP, la télédétection est assurée par des satellites, des avions, des ballons et des hélicoptères, des petits engins aériens sans pilote (UAS) ou drones et toute une variété de capteurs, tels que des capteurs optiques ou quasi-infrarouge, ou encore des radars.

En raison de leur coût peu élevé, les drones pourraient remplacer avantageusement les satellites et les avions. Les agriculteurs peuvent en effet les utiliser pour l'épandage de pesticides, pour suivre les déplacements du cheptel ou assurer le suivi des récoltes. Parmi les autres applications possibles de la télédétection dans AP, citons l'imagerie des sols nus pour la délimitation des zones de gestion, la cartographie des adventices, la détection du stress azoté, la cartographie des rendements et la détection des nuisibles et des maladies.[16]

1.5 Avantages et impacts de l'agriculture de précision

Les avantages de l'AP sont nombreux, qu'il s'agisse de rentabilité, de productivité, de développement durable, de qualité des cultures, de protection de l'environnement, de qualité de vie des agriculteurs, de sécurité alimentaire ou de développement économique des zones rurales. L'AP peut en effet accroître les rendements des cultures et améliorer ainsi la sécurité alimentaire. Les outils de l'AP peuvent aider les agriculteurs à économiser sur les coûts en augmentant l'efficacité des systèmes de production à grande échelle et améliorer les rendements et la rentabilité des exploitations agricoles grâce à une meilleure gestion des intrants.

Alors que les nuisibles et les maladies sont à l'origine d'importantes pertes agricoles dans les pays ACP, la télédétection peut aider à identifier des zones à très petite échelle contaminées par des pathogènes. L'épandage de fongicides peut ainsi se faire au meilleur moment. En

outre, lorsqu'elle est associée aux technologies GPS et SIG, la télédétection peut être utilisée dans le contrôle des mauvaises herbes spécifiques à un site. [16]

L'agriculture de précision a aussi un impact positif sur l'environnement puisqu'elle permet une utilisation plus ciblée des intrants, ce qui limite les pertes liées à l'application de doses excessives ou aux déséquilibres en nutriments, aux mauvaises herbes ou encore aux dommages provoqués par des insectes, par exemple. Des études ont en effet montré qu'une gestion des nutriments tenant compte des spécificités du site a permis de limiter l'utilisation d'engrais azotés au Vietnam et aux Philippines, de respectivement 14 % et 10 %. Les pertes d'azote au niveau du sol ont également diminué, de 25-25 %. L'épandage d'herbicides à débit variable a quant à lui réduit l'utilisation totale des herbicides et préservé ainsi la qualité des eaux de surface et souterraines. La contamination des sols et de l'eau s'en trouve ainsi minimisée.

L'agriculture de précision a également un impact au niveau de la société. Elle crée en effet des emplois, dans le domaine des technologies (matériel et logiciels informatiques, guidage d'engins, capteurs au niveau des sols et des cultures, gestion de l'information, systèmes d'aide à la décision) et limite la pollution environnementale provoquée par l'épandage de quantités excessives d'engrais. [2]

2 Les drones et leurs relations avec l'agriculture de précision (AP).

2.1 Les drones

Un drone est un aéronef sans passager ni pilote qui peut voler de façon autonome ou être contrôlé à distance depuis le sol. Le mot « drone » est une extrapolation d'un terme anglais qui signifie « faux-bourdon ». En français, le terme est employé pour désigner des véhicules aériens, terrestres, de surface ou sous-marins, alors que la classification anglo-saxonne distingue chaque type d'appareil.

La taille d'un drone aérien peut aller de quelques centimètres pour les modèles miniatures à plusieurs mètres pour les drones spécialisés (surveillance, renseignement, combat, transport, loisirs). L'autonomie en vol va de quelques minutes à plus de 40 heures pour les drones de longue endurance.[17]

2.2 Composants d'un drone [13] :

Lorsque l'on se penche de plus près, on remarque qu'un drone est composé de trois principaux composants qui sont :

- **Un châssis**

C'est la base du drone. Selon les modèles, le châssis peut être différent et être composé de 3, 4, 6 ou 8 bras. Les bras d'un drone peuvent être simples ou doubles. Le châssis d'un drone doit être ergonomique afin de permettre à l'objet de s'envoler facilement. Les châssis peuvent être composés de bois, d'aluminium, de plastique ou encore de fibre de carbone.

- **Un système de propulsion**

C'est ce qui va permettre au châssis de s'envoler. Pour cela, le système de propulsion comprend des moteurs, des hélices, des batteries et des contrôleurs de vitesse électroniques (ESC). Selon ces différents éléments, le temps de vol ainsi que la charge maximale supportée par le drone pourront différer.

- **Un contrôleur de vol**

Grâce à un microprocesseur et des capteurs, il permet de faire le lien entre le pilote et le drone. Les pièces détachées d'un drone peuvent être achetées séparément afin que les plus connaisseurs puissent fabriquer leur propre drone (DIY). Selon les fonctions du drone, d'autres composants peuvent être ajoutés comme une caméra embarquée. Il est même possible de joindre différentes innovations technologiques pour construire son drone en utilisant l'impression 3D pour créer le châssis.



Figure 2: Image représentative d'un drone

2.3 Les capacités du drone [13] :

▪ **Altitude maximale**

Les drones sont soumis à des restrictions, et ce, notamment concernant l'altitude maximale à ne pas dépasser. En effet, les pilotes de drone sont tenus de respecter **une altitude maximale de 150 m**. Au-delà, les drones rentreraient dans l'espace aérien et cela pourrait représenter un danger.

▪ **Vitesse de pointe**

La vitesse d'un drone dépend de son modèle. En effet, un drone ayant pour fonction principale la prise de vue aérienne n'ira pas aussi vite qu'un drone de course. En moyenne, les drones civils affichent **entre 20 et 80 km/h** au compteur, mais cela peut aller bien au-delà.

▪ **La capacité d'emport**

Utile pour capturer des vues aériennes, le drone est en passe de devenir un moyen de livraison très intéressant. Cependant, pour pouvoir livrer, les drones doivent pouvoir porter une masse assez importante. C'est ce qu'on appelle la capacité d'emport. Quel poids peut donc porter un drone ? Si aujourd'hui la majeure partie des drones civils ont une capacité d'emport de quelques kilos maximum, certains bijoux de technologie très puissants sont capables de supporter une masse beaucoup plus importante comme le drone Megacopter ayant supporté une masse de 61 kg en vol pendant 37 secondes !

▪ **Talon d'Achille**

Le seul véritable ennemi du drone reste la météo et plus particulièrement le vent. La problématique du vent pour les drones peut amener à une perte de contrôle.

2.4 Typologie de drones [18] :

Il existe en effet plusieurs modèles de drones pour répondre à des besoins très divers :

✓ **Le drone loisir ou mini drone**

Il prend de plus en plus de place dans le rayon jouet des enfants, mais correspond également aux adultes débutants souhaitant s'amuser. Certains modèles loisirs sont dotés d'une caméra pour la prise de vue aérienne. Ce type de drone a l'avantage d'être peu onéreux et se pilote

facilement. On retrouve de plus en plus de mini drones pouvant être pilotés en intérieur et parfois même en extérieur.

✓ **Le drone avec caméra**

Ce type de drone s'adresse aux personnes souhaitant réaliser de l'imagerie aérienne. Les caméras sont montées sur des axes pour une grande stabilité et selon les modèles, les caméras embarquées peuvent filmer en HD, full HD, 4 K voire même davantage pour les drones destinés au cinéma.

✓ **Le drone de course**

Le drone de course permet de pratiquer que le FPV (First Person View) Racing, c'est-à-dire une course d'obstacles en totale immersion avec des figures acrobatiques. Les images capturées par la caméra du drone diffusent en effet en temps réel le parcours au pilote muni de lunettes FPV ou d'un masque FPV. Le vol en immersion demande de nombreuses heures d'entraînement et se pratique généralement sur des terrains privés.

✓ **Le drone professionnel**

Tout drone de plus de 800 g est considéré comme un drone professionnel et doit être enregistré à la DGAC (Direction Générale de l'Aviation Civile). Les drones professionnels ont de multiples fonctions et sont plus complexes à piloter. Les pilotes professionnels peuvent survoler de nuit ou dans les zones urbaines s'ils en font la demande et obtiennent des autorisations. Les drones professionnels servent dans de nombreux domaines : thermographie, cartographie, surveillance, contrôle, recherche, etc.

✓ **Le drone militaire :**

À l'origine, les drones aériens ont été développés pour des usages militaires, principalement pour des missions de reconnaissance, de surveillance et d'attaques ciblées.

Les drones sont plus économiques tout en évitant de mettre en jeu la vie des pilotes et de déployer des troupes terrestres notamment pour les missions de reconnaissance, de surveillance et les attaques ciblées. Leur utilisation au sein des armées et forces de police est devenue prépondérante.

2.5 Les fonctions du drone [13] :

❖ Un outil de journalise

Le drone est utilisé dans ce cas pour tourner des documentaires et des films. Il est en effet utilisé dans le cadre du journalisme pour **atteindre des zones sinistrées** ou difficilement accessibles. Cela permet de réaliser des reportages sans mettre en danger les journalistes. Il permet aussi d'avoir des plans aériens sur de grandes manifestations sans avoir à utiliser un hélicoptère (ce qui est bien plus coûteux)

❖ Des plans cinématographiques

Les drones sont également de plus en plus utilisés au cinéma embarquant des caméras de très haute qualité. Grâce aux stabilisateurs, les images capturées par les drones dédiés au cinéma sont nettes. Ils permettent de réaliser des plans tout à fait remarquables et sont **utilisés par de nombreux réalisateurs** pour remplacer l'hélicoptère, la grue ou le steady cam. Les drones sont également parfois utilisés pour réaliser des **clips musicaux ou des vidéos YouTube**.

❖ La promotion du territoire

De nombreux **offices de tourisme** utilisent aujourd'hui le drone pour réaliser des plans de la ville ou du territoire qu'il souhaite promouvoir. Quoi de plus beau que des vues aériennes sur les monuments et sites remarquables d'une région pour donner envie aux touristes de se rendre sur place.

❖ Un outil pour l'agriculture

C'est peut-être la fonction la plus inattendue du drone et pourtant le drone peut être un appareil véritablement utile pour les agriculteurs souhaitant surveiller leurs champs. Outre la surveillance, les drones spécialisés dans le domaine de l'agriculture sont capables de **fournir des services très pointus** : repérage des hétérogénéités du sol, examen de la densité végétale, création de zones sur les parcelles, etc.

❖ De nombreuses autres fonctions en développement

Les drones n'ont pas fini de nous surprendre et de nous simplifier la vie au quotidien. Ainsi, il est très probable que dans quelques années ses fonctions évoluent encore et proposent des services de livraison (pour Amazon, Domino's ou encore Walmart), des services de modélisation 3D, mais pourront également venir de plus en plus épauler les services de police et de secours.

2.6 Les drones au service de l'agriculture

Pour réaliser des prestations techniques d'agriculture de précision, le professionnel doit être équipé de drones multi-rotors ou d'ailes volantes, pouvant emporter différents types de caméras, appareils photographiques, capteurs spécifiques de types RGB ou multi spectral pour réaliser des cartographies NDVI (Normalized Difference Vegetation Index) indice de végétation par différence normalisée, de type PRI-R (Proche infrarouge, Rouge) ou RVB (Rouge, Vert, Bleu, longueur d'onde visible).

Ces cartographies NDVI sont générées par des logiciels informatiques très puissants, donnant des images techniques extrêmement précises, de contrôler et analyser tous types de cultures agricoles, blé, orges, betteraves, colza, lins, maïs, tournesols, etc. Permettant aux agriculteurs et ingénieurs agronomes d'établir des diagnostics sur la vitalité et la santé des cultures et apporter des solutions techniques.

L'aéronef utilisé sur des petites parcelles ou sur des exploitations de plusieurs hectares, survol à très basse altitude les plantations lui permettent de prendre des images et mesures d'une grande précision, le survol étant géo localisée par un GPS permet de pouvoir localiser les zones où les végétaux ont besoin d'une attention particulière de la part de l'agriculteur.

L'étude est l'analyse des parcelles agricoles par cartographie NDVI est un outil d'aide à la prise de décision dans l'agriculture, en donnant des informations sur la vigueur des végétaux et l'état phytosanitaire des plantations, pour définir précisément les zones à traiter ou irriguer, optimisant l'apport d'engrais. Ces études permettent aussi de repérer les zones touchées par la maladie pour cibler les zones à traiter évitant ainsi le traitement des zones cultivées saines.

Cette technique permet donc aux agriculteurs de mieux gérer les cultures, d'augmenter les rendements et de faire des économies considérables en limitant l'utilisation des traitements, préservent ainsi la planète.[2]

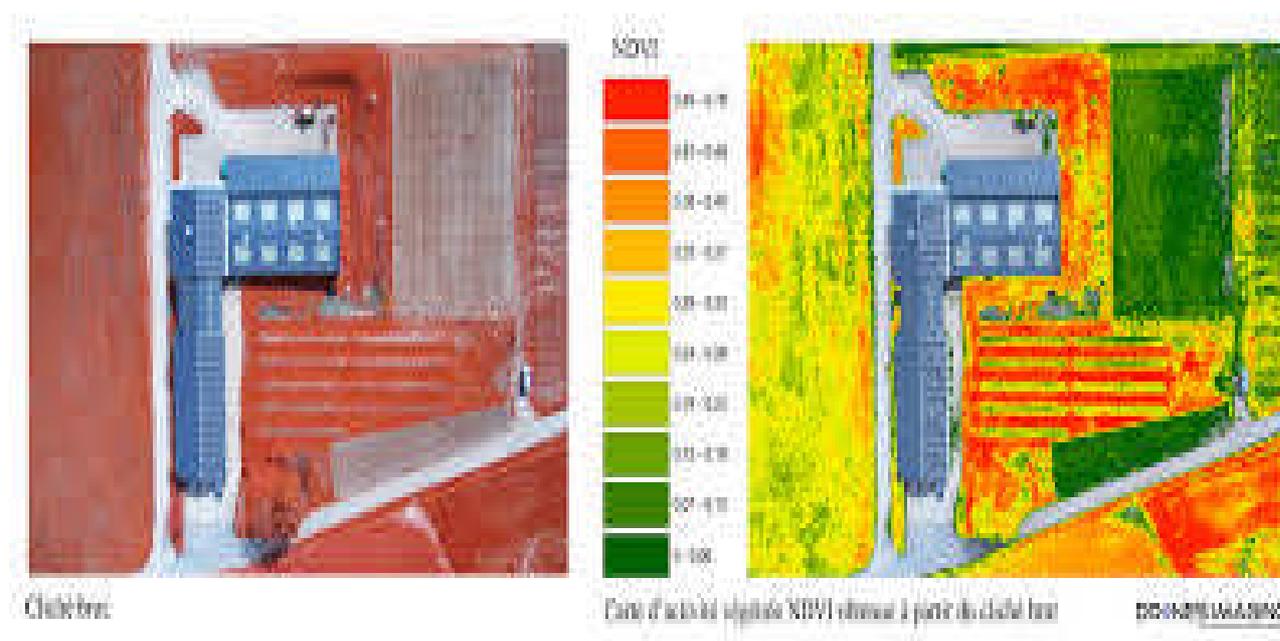


Figure 3: Images techniques réalisées par un drone.

2.7 La cartographie NDVI, un outil d'une extrême efficacité

L'agriculture de précision requiert l'utilisation d'un drone équipé de capteurs multi spectraux. Grâce aux images prises par leurs capteurs, les aéronefs fournissent des indicateurs agronomiques sans avoir à effectuer de prélèvements. En survolant une parcelle, le drone enregistre donc une multitude d'images géolocalisées, ce qui valorise leur exploitation de façon optimale. Après la récolte des données brutes prises lors du survol, ces informations sont analysées et interprétées par de puissants logiciels, pour produire une cartographie détaillée d'une extrême précision, offrant aux ingénieurs agronomes les ressources nécessaires à l'établissement des diagnostics et préconisations visant à préserver la santé et la vitalité des cultures

La cartographie NDVI est un outil d'aide à la prise de décision pour l'agriculteur. Elle fournit de précieux renseignements sur l'état phytosanitaire des parcelles et favorise la diminution des coûts liés à l'exploitation en ciblant précisément les zones en manques d'irrigations, les zones sinistrées nécessitant intervention et traitement. En cartographiant des parcelles agricoles mesurant jusqu'à plusieurs hectares, cette technique permet aux exploitants de connaître les surfaces exactes des parcelles mais également d'inspecter, contrôler, diagnostiquer et améliorer la gestion des cultures, optimiser les récoltes, augmenter les rendements et de faire des économies considérables en évaluant les doses d'intrants selon les

zones de votre parcelle et en limitant l'utilisation de produits ainsi que la quantité d'engrais utilisé. [12]

2.8 Méthodes d'optimisation et réduction [12] :

Grace aux drones, on peut optimiser les rendements agricoles et réduire les couts. Dans ce contexte, les usages les plus fréquents sont :

- **Le repérage**

Grâce à sa fonction GPS, le drone survole la parcelle pour réaliser un ensemble de prises de vues aériennes selon un itinéraire programmé par le télé pilote. le professionnel doit être équipé de drones multi-rotors ou d'ailes volantes, embarque différents types de caméras et capteurs spécifiques (*de types RGB ou multi spectral*) pour réaliser des cartographies NDVI .Il couvre de grandes surfaces en un temps record et fournit aux ingénieur agronomes, des informations indispensables sur les parcelles et les cultures qui s'y trouvent.

- **La pulvérisation**

Équipé d'un réservoir et de buses de pulvérisation, le drone survole les parcelles et pulvérise différents types de produits avec une précision et une rapidité qui démontrent son efficacité en comparaison au traditionnel procédé manuel. L'aéronef couvre de façon optimale des surfaces plus ou moins grandes tout en évitant de piétiner les cultures au sol ou leur écrasement par les roues d'un véhicule agricole.



Figure 4: Photo d'un drone pulvérisateur

Conclusion

Dans ce chapitre, on a vu qu'est-ce que c'est que l'agriculture de précision et son impact dans le monde moderne, on a introduit la notion de drone et ses typologies et on a vu l'utilité et la place des drones dans l'agriculture de précision. Dans le prochain chapitre on verra les méthodes et approches de la classification d'images on va s'intéresser aux images agricoles et voir le principe de l'apprentissage profond (Deep Learning) ainsi que les réseaux de neurones convolutifs (CNNs).

CHAPITRE 02 :

**Deep Learning ,Machine Learning ,Réseaux
de neurones et Classification d'images.**

Introduction :

L'intelligence artificielle (IA) a bien connu son essor ces dernières années, on la voit s'infiltrer de plus en plus dans la vie quotidienne des usagers des appareils numériques, il n'y a presque plus aucun domaine où l'on ne peut pas la trouver.

Le Deep Learning est un nouveau domaine de recherche du ML, qui a été introduit dans le but de rapprocher le ML de son objectif principal : l'intelligence artificielle. Il concerne les algorithmes (réseaux de neurones) inspirés par la structure et le fonctionnement du cerveau. Ils peuvent apprendre plusieurs niveaux de représentation dans le but de modéliser des relations complexes entre les données.

La classification automatique des images consiste à attribuer automatiquement une classe à une image à l'aide d'un système de classification. On retrouve ainsi la classification d'objets, de scènes, de textures, la reconnaissance de visages, d'empreintes digitales et de caractères. Il existe deux principaux types d'apprentissage : l'apprentissage supervisé et l'apprentissage non-supervisé. Dans l'approche supervisée, chaque image est associée à une étiquette qui décrit sa classe d'appartenance. Dans l'approche non supervisée les données disponibles ne possèdent pas d'étiquettes. Dans notre travail on s'intéresse à l'approche supervisée.

1 . Deep Learning, Machine Learning et les réseaux de neurones :

1.1 Principe d'Apprentissage Automatique (Machine Learning) :

L'apprentissage automatique (Machine Learning [ML] en anglais), qui est un sous domaine de l'intelligence artificielle, 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 voire impossible de les remplir par des moyens algorithmiques classiques.

Pour être plus clair, ce que fait le machine learning, c'est apprendre à résoudre un problème de manière automatique en utilisant les données. Dans le cadre du traitement de la donnée, il s'agit de construire un modèle obtenu directement à partir d'exemples. L'objectif des algorithmes de machine learning est de minimiser ce qu'on appelle l'erreur, c'est-à-dire de se tromper le moins possible .Les méthodes d'apprentissage automatique les plus adoptées sont l'apprentissage supervisé et l'apprentissage non supervisé. [1]

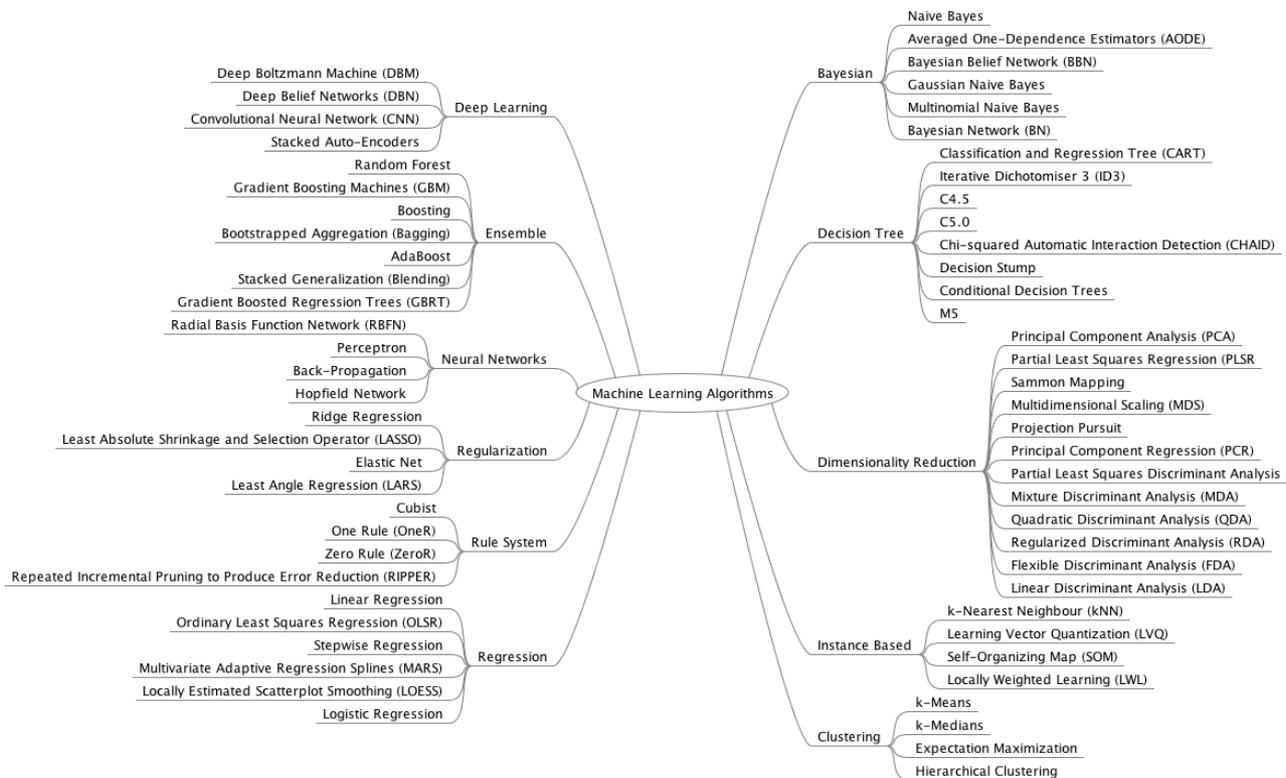


Figure 5: Les différents algorithmes de Machine Learning

1.2. La notion d'apprentissage :

Quand avons-nous besoin de l'apprentissage automatique ?

Quand avons-nous besoin de l'apprentissage automatique plutôt que de programmer directement nos ordinateurs pour effectuer la tâche à accomplir ? Deux aspects d'un problème donné peuvent nécessiter l'utilisation de programmes qui apprennent et s'améliorent sur la base de leur expérience.

- Tâches trop complexes à programmer (tâches accomplies par les humains): Il y a de nombreuses tâches que nous, humains, accomplissons régulièrement, mais notre introspection concernant la façon dont nous les faisons n'est pas suffisamment élaborée pour extraire un programme bien défini. Des exemples de telles tâches comprennent la conduite, la reconnaissance de la parole et la compréhension de l'image. Dans toutes ces tâches, des programmes d'apprentissage automatique à la fine pointe de la technologie, des programmes qui tirent des leçons de leur expérience, « obtiennent des résultats tout à fait satisfaisants, une fois exposés à suffisamment d'exemples de formation.
- Tâches au-delà des capacités humaines: une autre grande famille de tâches bénéficiant des techniques d'apprentissage automatique est liée à l'analyse d'ensembles de données très volumineux et complexes: données astronomiques, transformation des archives médicales en connaissances médicales, prédiction météorologique, analyse de données génomiques, il devient évident qu'il existe des trésors d'informations significatives enfouies dans des archives de données qui sont beaucoup trop grandes et trop complexes pour que les humains aient du sens. Apprendre à détecter des modèles significatifs dans des ensembles de données volumineux et complexes est un domaine prometteur dans lequel la combinaison de programmes qui apprennent avec la capacité de mémoire presque illimitée et la vitesse de traitement croissante des ordinateurs ouvre de nouveaux horizons.

1.3. Définition et Principe de l'Apprentissage Profond (Deep Learning) :

Le Deep Learning ou apprentissage profond est un type d'intelligence artificielle dérivé du machine learning (apprentissage automatique) où la machine est capable d'apprendre par elle-même, contrairement à la programmation où elle se contente d'exécuter à la lettre des règles prédéterminées.

Dans l'apprentissage automatique, un programme analyse un ensemble de données afin de tirer des règles qui permettront de tirer des conclusions sur de nouvelles données

Le deep Learning s'appuie sur un réseau de neurones artificiels s'inspirant du fonctionnement des neurones biologiques du cerveau humain. Ce réseau est composé de dizaines voire de centaines de « couches » de neurones, chacune recevant et interprétant les informations de la couche précédente.

Voici une figure qui nous montre la relation entre les trois (3) concepts cités , Intelligence Artificielle (AI), Machine Learning (ML) et Deep Learning. [4]

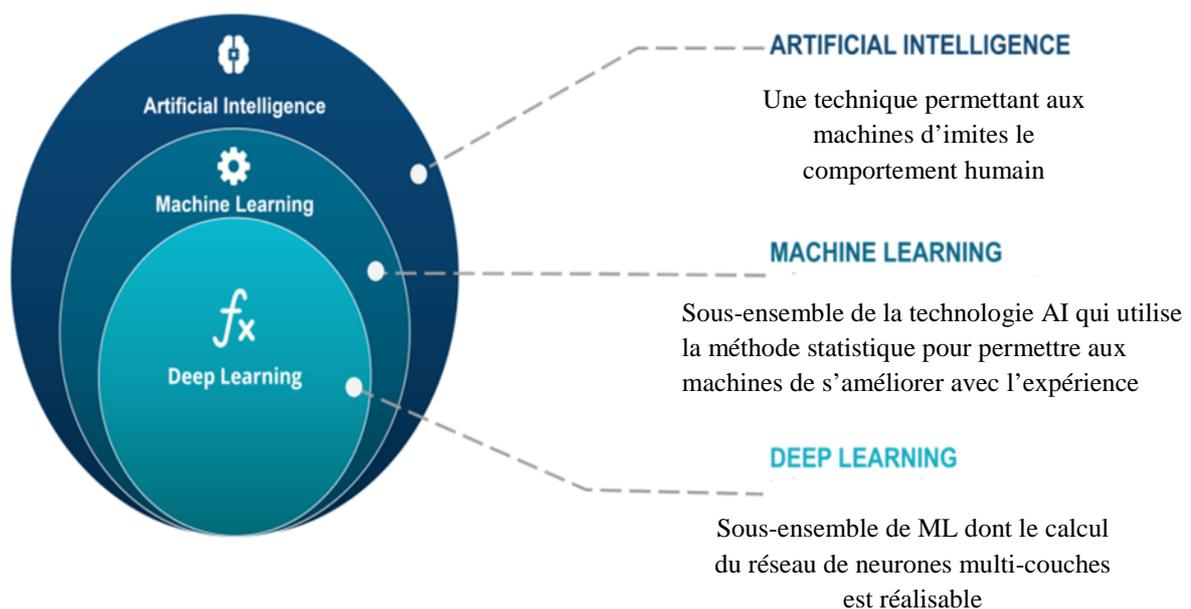


Figure 6 : La relation entre l'IA, ML et le Deep Learning.

1.4. Historique de Deep Learning :

Tout a commencé à partir d'une graine, une graine qui s'est développée et qui n'a cessé de grandir pour créer le DeepLearning.

Le deep learning est un concept nouveau qui émerge depuis les années 2000. Si le deep learning lui est nouveau, ce n'est pas le cas des réseaux de neurones artificiels, concept sur lequel se base le deep Learning. On entend parler du premier neurone artificiel en 1943 lorsque Warren McCulloch et Walter Pitts publient leur premier modèle mathématique et informatique du neurone biologique: le neurone formel.

Le neurone formel est directement inspiré du neurone biologique.

On a les différentes phases à par lesquelles le deeplearning s'est développé :

- 1957 Perceptron : Le perceptron est le réseau de neurone le plus simple, il est composé de neurones qui fonctionnent légèrement différemment que le neurone formel.
- 1986 MLP Les perceptrons multicouches : les perceptrons multicouches ont pour objectif de classer différentes données selon leur étiquette. Pour cela le perceptron observe chacune des données qu'il possède et met à jour chaque poids de chaque neurone de son réseau afin de classifier au mieux cette base de données. L'algorithme que les perceptrons utilisent pour mettre à jour leurs poids s'appelle la rétro propagation du gradient de l'erreur.
- 1992 SVM Les machines à vecteurs de support : sont un ensemble de techniques d'apprentissage supervisé destinées à résoudre des problèmes de discrimination et de régression. Les SVM sont une généralisation des classifieurs linéaires. [14]
- 2010 Deep Neural Networks : que l'on appelle un réseau de neurones profond est un perceptron avec au minimum deux couches cachées (il peut y en avoir autant qu'on le souhaite).

La révolution du deep learning est liée à la puissance des ordinateurs qui ne cesse d'augmenter. L'essor du deep learning vient également de la quantité de données qui ne cesse de s'accumuler. Au fur et à mesure que les RNA devenaient de plus en plus puissants et complexes - et plus profonds avec de nombreuses couches et de nombreux neurones -, la capacité d'apprentissage en profondeur facilitait l'apprentissage machine robuste et augmentait l'IA. [15]

1.5. Pourquoi le Deep Learning ?

Les algorithmes de ML décrits dans la première partie fonctionnent bien pour une grande variété de problèmes. Cependant ils ont échoué à résoudre quelques problèmes majeurs de l'IA telle que la reconnaissance vocale et la reconnaissance d'objets. Le développement du Deep Learning fut motivé en partie par l'échec des algorithmes traditionnels dans de telle tâche de l'IA. Mais ce n'est qu'après que de plus grandes quantités de données ne soit disponibles grâce notamment au Big Data et aux objets connectés et que les machines de calcul soient devenues plus puissantes qu'on a pu comprendre le potentiel réel du Deep Learning.

Une des grandes différences entre le Deep Learning et les algorithmes de ML traditionnelles c'est qu'il s'adapte bien, plus la quantité de données fournie est grande plus les performances d'un algorithme de Deep Learning sont meilleures. Contrairement à plusieurs algorithmes de ML classiques qui possèdent une borne supérieure à la quantité de données qu'ils peuvent recevoir des fois appelée "plateau de performance", les modèles de Deep Learning n'ont pas de telles limitations (théoriquement) et ils sont même allés jusqu'à dépasser la performance humaine dans des domaines comme l'image processing.[3]

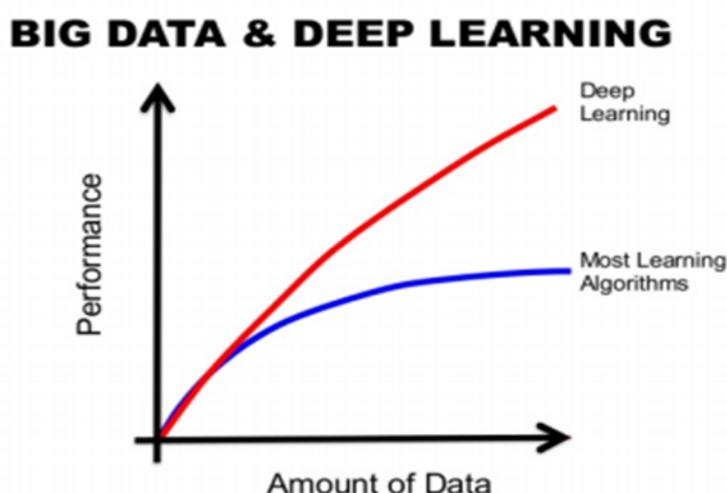


Figure 7: La différence de performance entre le Deep Learning et la plupart des algorithmes de ML en fonction de la quantité de données

Autre différence entre les algorithmes de ML traditionnelles et les algorithmes de Deep Learning c'est l'étape de l'extraction de caractéristiques. Dans les algorithmes de ML traditionnelles l'extraction de caractéristiques est faite manuellement, c'est une étape difficile et coûteuse en temps et requiert un spécialiste en la matière alors qu'en Deep Learning cette étape est exécutée automatiquement par l'algorithme.[19]

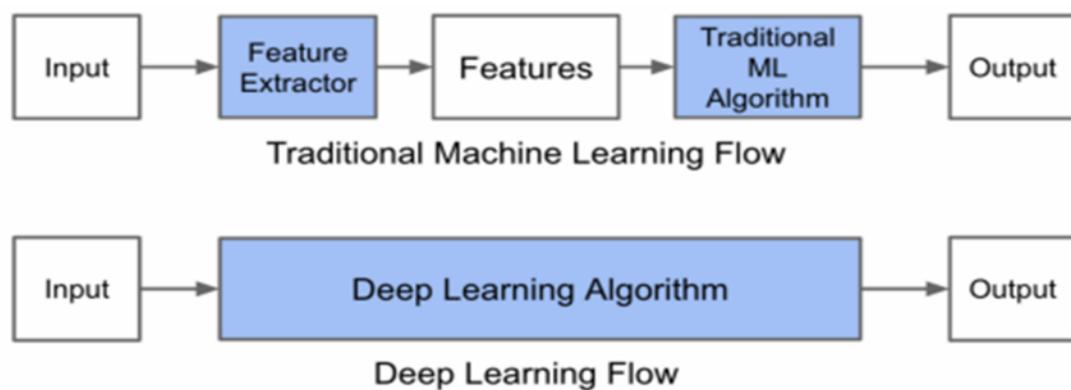


Figure 8 :Le procédé du ML classique comparé à celui du Deep Learning

1.6. Fonctionnement du Deep Learning

Au sein du cerveau humain, chaque neurone reçoit environ 100 000 signaux électriques des autres neurones. Chaque neurone en activité peut produire un effet excitant ou inhibiteur sur ceux auxquels il est connecté. Au sein d'un réseau artificiel, le principe est similaire. Les signaux voyagent entre les neurones. Toutefois, au lieu d'un signal électrique, le réseau de neurones assigne un certain poids à différents neurones. Un neurone qui reçoit plus de charge exercera plus d'effet sur les neurones adjacents. La couche finale de neurones émet une réponse à ces signaux.[5]

1.7. Domaine d'applications du *Deep Learning* [5] :

Le Deep Learning a permis d'obtenir des résultats impressionnants dans des domaines aussi nombreux que variés :

- Reconnaissance d'image,
- Traduction automatique,
- Voiture autonome,
- Diagnostic médical,
- Recommandations personnalisées,
- Modération automatique des réseaux sociaux,
- Prédiction financière et trading automatisé,
- Identification de pièces défectueuses,

- Détection de malwares ou de fraudes,
- Exploration spatiale,
- Robots intelligents.

On pourrait citer encore beaucoup d'exemple plus originaux les uns que les autres, mais ce qu'il faut retenir c'est que le Deep Learning permet de faire apprendre à un ordinateur une tâche précise en observant un grand nombre d'exemples.

1.8. Les Réseaux de Neurones

Dans ce titre, nous parlons sur les réseaux de neurones en montrant leurs principes et typologies. Dans le prochain titre, nous détaillons les réseaux neurones convolutionnels et la raison qui nous a poussé à choisir cette typologie.

1.8.1. Modélisation d'un neurone : une inspiration biologique :

Dans les réseaux de neurones, l'élément de base est un neurone artificiel qui correspond à un modèle issu des observations faites sur les neurones biologiques.

Le neurone biologique est une cellule excitable, Il reçoit et traite en permanence des informations en provenance des dendrites, des stimuli qui vont agir sur la répartition des ions de part et d'autre de sa membrane. Lorsque la somme de ces courants électriques atteint un certain seuil, l'équilibre électrochimique est modifié localement et un influx nerveux (un courant électrique) est déclenché, se propage le long de l'axone puis atteint les neurones voisins reliés par les synapses. On parle de potentiel d'activation.

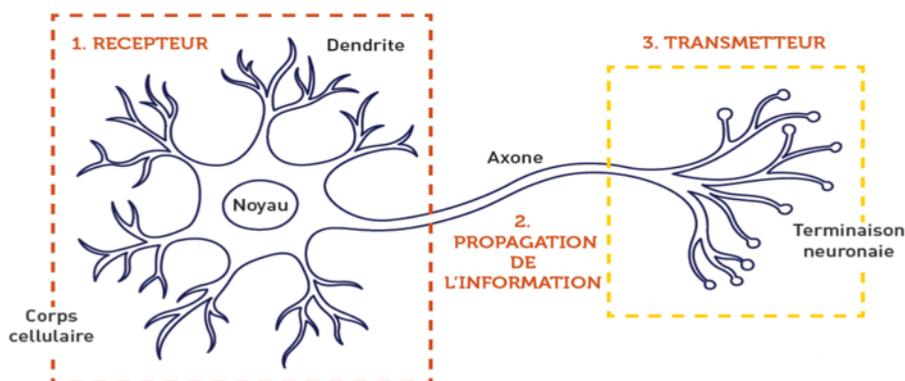


Figure 9: Schémas d'un neurone biologique.

C'est de ce mécanisme dont est inspiré le neurone artificiel, ou neurone formel, modélisé initialement par Warren McCulloch et Walter Pitts en 1943. Dans sa forme la plus simple, il peut se représenter de la façon suivante

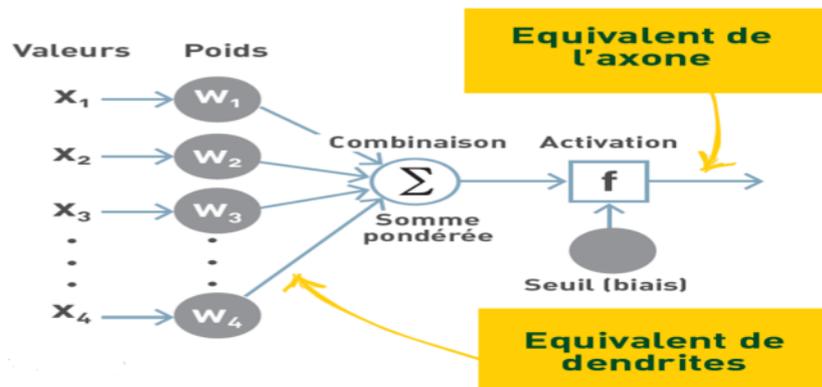


Figure 10: schémas d'un neurone formel.

Les valeurs (x_i) sont les données d'entrée (les variables que l'on souhaite prendre en compte). On peut les comparer aux messages transmis par les liaisons entre les neurones (dendrites). Ces messages arrivent au neurone de manière pondérée c'est-à-dire que, pour traduire leur importance pour la résolution du problème, un poids (w_i) leur est attribué (on parle de « poids synaptique »).

Le neurone combine d'abord ces informations en faisant la somme pondérée de tous les messages ($z = \sum w_i x_i$). Puis, une fonction mathématique f , basée sur cette somme pondérée, permet l'activation du neurone c'est-à-dire la production d'un message Y en sortie.

La fonction d'activation s'apparente au potentiel d'activation du neurone biologique. Il peut par exemple s'agir d'une fonction de seuil qui renvoie 1 si un seuil (w_0) est atteint et 0 si le seuil n'est pas atteint. $Y = f(z) = f(\sum w_i x_i + w_0)$

1.8.2. Principe des réseaux de neurones

Les réseaux de neurones proposent une simulation du fonctionnement de la cellule nerveuse à l'aide d'un automate : le neurone formel.

Les réseaux neuronaux sont constitués d'un ensemble de neurones (nœuds) connectés entre eux par des liens qui permettent de propager les signaux de neurone à neurone.

Grâce à leur capacité d'apprentissage, les réseaux neuronaux permettent de découvrir des relations complexes non-linéaires entre un grand nombre de variables, sans intervention externe. De ce fait, ils sont largement utilisés dans de nombreux problèmes de classification

(ciblage marketing, reconnaissance de formes, traitement de signal, ...) d'estimation (modélisation de phénomènes complexes, ...) et prévision (bourse, ventes, ...).

Il existe un compromis entre clarté du modèle et pouvoir prédictif. Plus un modèle est simple, plus il sera facile à comprendre, mais moins il sera capable de prendre en compte des dépendances trop variées.

1.8.3. Typologie des réseaux neuronaux :

Il existe beaucoup de types de réseaux neurones, chaque type étant développé pour un objectif particulier.

1.8.3.1. Neurone Formel

Un neurone formel est une représentation mathématique et informatique d'un neurone biologique. Le neurone formel possède généralement plusieurs entrées et une sortie qui correspondent respectivement aux dendrites et au cône d'émergence du neurone biologique (point de départ de l'axone). Les actions excitatrices et inhibitrices des synapses sont représentées, la plupart du temps, par des coefficients numériques (les poids synaptiques) associés aux entrées. Les valeurs numériques de ces coefficients sont ajustées dans une phase d'apprentissage.

Dans sa version la plus simple, un neurone formel calcule la somme pondérée des entrées reçues, puis applique à cette valeur une fonction d'activation, généralement non linéaire. La valeur finale obtenue est la sortie du neurone.

Le neurone formel est l'unité élémentaire des réseaux de neurones artificiels dans lesquels il est associé à ses semblables pour calculer des fonctions arbitrairement complexes, utilisées pour diverses applications en intelligence artificielle. [17]

1.8.3.2. Neurones multicouches

Le perceptron multicouche (*multi layer perceptron* MLP) est un classifieur linéaire de type réseau neuronal formel organisé en plusieurs couches au sein desquelles une information circule de la couche d'entrée vers la couche de sortie uniquement ; il s'agit donc d'un réseau de type **feedforward**. Chaque couche est constituée d'un nombre variable de neurones, les neurones de la couche de sortie correspondant toujours aux sorties du système.[17]

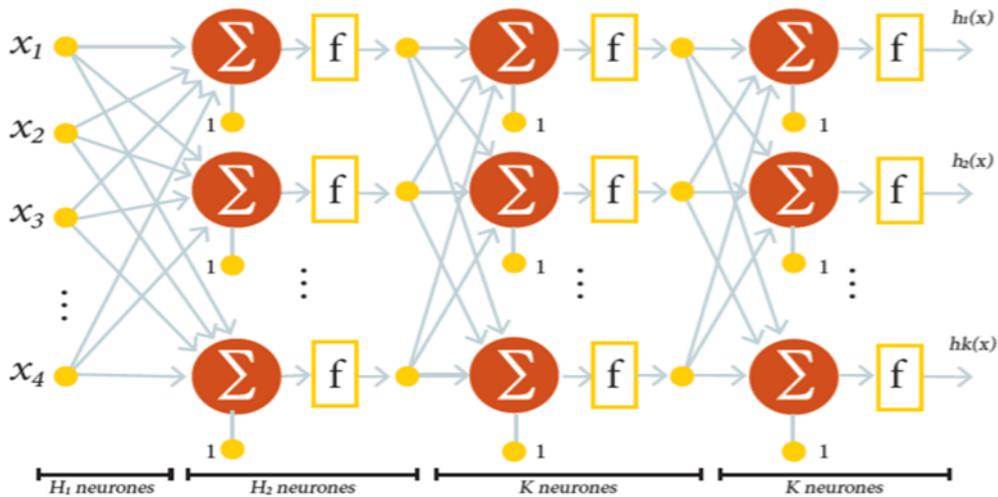


Figure 11: Représentation d'un perceptron multi-couches

1.8.3.3. Neurones récurrents

Les réseaux de Neurones récurrents (RNNs) permettent d'analyser les séquences de vecteurs tout comme les modèles de Markov cachés. Le temps entre ici en ligne de compte car les sorties (de la couche de sortie et/ou de la couche cachées) calculées à l'instant t sont réinjectées en entrée du réseau et/ou en entrée de la couche cachée. On peut en théorie conserver dans le réseau la mémoire de ce qui s'y est passé depuis le début.

Un réseau de neurones récurrent peut être considéré comme plusieurs copies du même réseau, chacune transmettant un message à son successeur. [3]

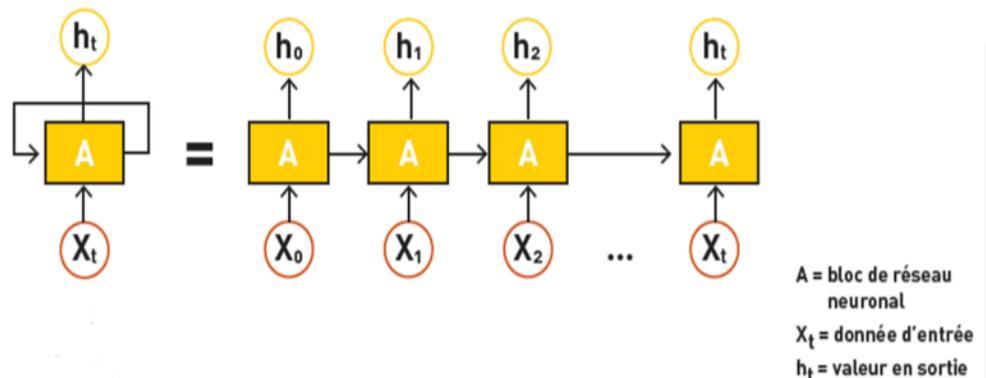


Figure 12: principe général d'un réseau e neurones récurrents.

1.8.3.4. Réseaux Neurones Convolutionnels (CNN)

En apprentissage automatique, un réseau de neurone convolutifs (ou réseau de neurones à convolution, ou CNN ou ConvNet) est un type de réseau de neurones artificiels acycliques (Feed-forward) dans lequel le motif de connexion entre les neurones est inspiré par le cortex visuel des animaux. Les neurones de cette région du cerveau sont arrangés de sorte à ce qu'ils correspondent à des régions qui se chevauchent lors du pavage du champ visuel. Leur fonctionnement est inspiré par les processus biologiques, ils consistent en un empilage multicouche de perceptrons, dont le but est de prétraiter de petites quantités d'informations.

Les réseaux neuronaux convolutifs ont de larges applications dans la reconnaissance d'image et vidéo, les systèmes de recommandation et le traitement du langage naturel.[3]

2. Réseaux de Neurones Convolutifs (CNNs) :

2.1. Principe d'architecture d'un CNN

Les réseaux de neurones convolutifs sont à ce jour les modèles les plus performants pour classer des images. Ils comportent deux parties bien distinctes une première partie que l'on appelle la partie convolutive du modèle et la seconde partie, que l'on va appeler la partie classification du modèle qui correspond à un modèle MLP (Multi Layers Perceptron).[3]

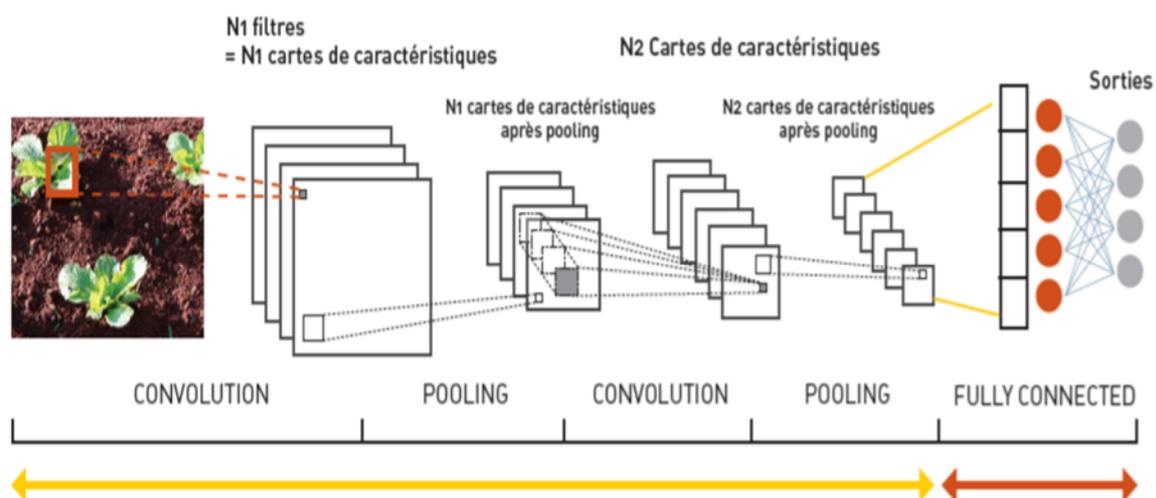


Figure 13: Architecture d'un réseau de neurones convolutifs.

2.2. Qu'est-ce que la convolution ?

La convolution est un outil mathématique simple qui est très largement utilisé pour le traitement d'image, ce qui explique que les réseaux de neurones à convolution soient particulièrement bien adaptés à la reconnaissance d'image.

La convolution agit comme un filtrage. On définit une taille de fenêtre qui va se balader à travers toute l'image (rappelez-vous qu'une image peut être vue comme étant un tableau). Au tout début de la convolution, la fenêtre sera positionnée tout en haut à gauche de l'image puis elle va se décaler d'un certain nombre de cases (c'est ce que l'on appelle le pas) vers la droite et lorsqu'elle arrivera au bout de l'image, elle se décalera d'un pas vers le bas ainsi de suite jusqu'à ce que le filtre ait parcouru la totalité de l'image.[15]

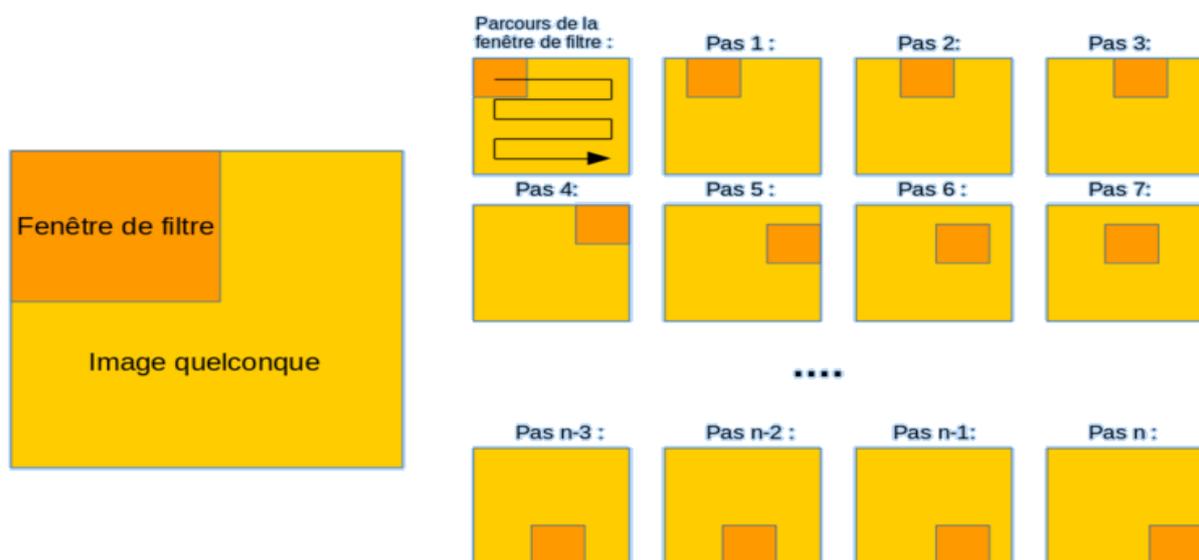


Figure 14: Schéma du parcours de la fenêtre de filtre sur l'image.

Le but est de se servir des valeurs présentes dans le filtre à chaque pas. La convolution va effectuer une opération avec ces pixels. Il peut s'agir de n'importe quelle opération, par exemple on extrait la valeur la plus grande (soit le pixel avec la plus grande valeur).

Exemple de convolution

Voici une convolution dont la configuration est :

- Opération = Argument maximale,
- Pas horizontale = 1 pixel,
- Pas vertical = 1 pixel.

On fait glisser la fenêtre en orange et à chaque pas on récupère la valeur la plus grande parmi les 9 valeurs de pixels. On remarque que la sortie de la convolution, que l'on peut appeler « carte de caractéristiques », à des dimensions plus petites que celle de l'image en entrée.

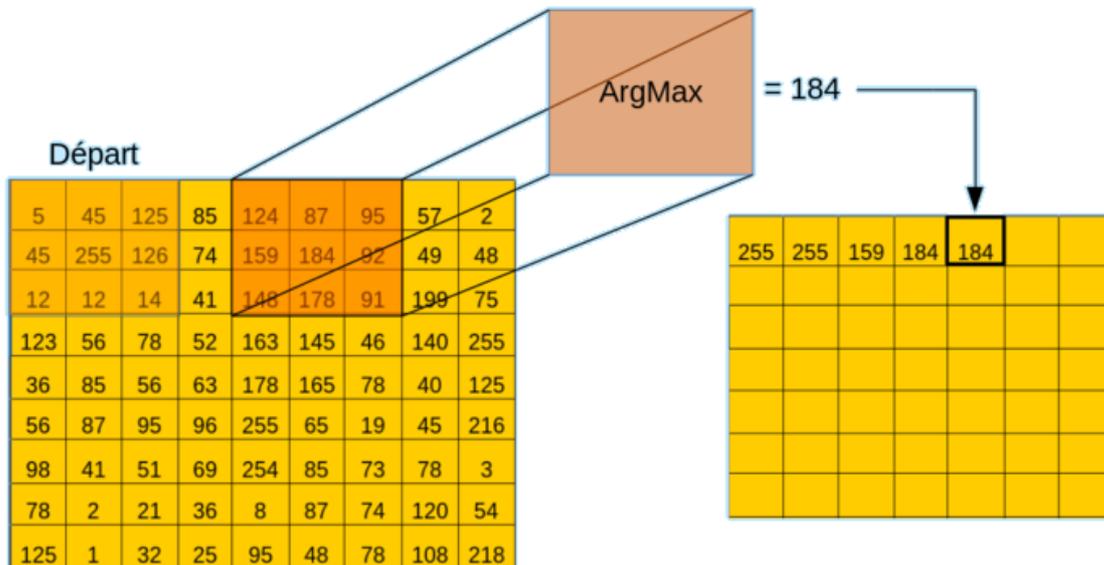


Figure 15: Exemple d'une convolution.

3. Les blocs de construction d'un apprentissage en profondeur

Une architecture CNN est formée par un empilement de couches de traitement indépendantes et entièrement connectées, il existe plusieurs types de couches et leur nombre dépendent de l'application ou des données particulières.

Exemple, un réseau plus petit avec seulement une ou deux couches de convolution peut être suffisant pour apprendre sur un petit nombre de données d'image en niveaux de gris. D'un autre côté, pour des données plus complexes avec des millions d'images colorées, vous aurez peut-être besoin d'un réseau plus complexe avec plusieurs couches convolutives et entièrement connectées.

3.1. Couche d'image d'entrée (Image Input Layer)

Image Input Layer définit la taille des images d'entrée d'un réseau de neurones convolutives et contient les valeurs de pixels brutes des images. On peut spécifier la taille de l'image à l'aide

de l'argument input Size et elle correspond à la hauteur, la largeur et le nombre de canaux de couleur de cette image, donc pour une image en niveaux de gris, le nombre de canaux est 1, et pour une image couleur (RGB), il est 3.

3.2. La couche de convolution (convolutionnet Layer) [20] :

La **couche de convolution** est la composante clé des réseaux de neurones convolutifs, et constitue toujours au moins leur première couche.

Son but est de repérer la présence d'un ensemble de *features* dans les images reçues en entrée. Pour cela, on réalise un filtrage par convolution : le principe est de faire "glisser" une fenêtre représentant la *feature* sur l'image, et de calculer le produit de convolution entre la *feature* et chaque portion de l'image balayée. Une *feature* est alors vue comme un filtre : les deux termes sont équivalents dans ce contexte.

La couche de convolution reçoit donc en entrée plusieurs images, et calcule la convolution de chacune d'entre elles avec chaque filtre

Contrairement aux méthodes traditionnelles, les *features* ne sont pas pré-définies selon un formalisme particulier, mais apprises par le réseau lors la phase d'entraînement. Les noyaux des filtres désignent les poids de la couche de convolution. Ils sont initialisés puis mis à jour par rétropropagation du gradient.

C'est là toute la force des réseaux de neurones convolutifs : ceux-ci sont capables de déterminer tout seul les éléments discriminants d'une image, en s'adaptant au problème posé. Par exemple, si la question est de distinguer les chats des chiens, les *features* automatiquement définies peuvent décrire la forme des oreilles ou des pattes.

Trois hyper paramètres permettent de dimensionner le volume de la couche de convolution (Aussi appelé volume de sortie) : la 'profondeur', le 'pas' et la 'marge'.

- **'Profondeur' de la couche**

Nombre de noyaux de convolution (ou nombre de neurones associés à un même champ récepteur).

- **'Le pas'**

Contrôle le chevauchement des champs récepteurs. Plus le pas est petit, plus les champs récepteurs se chevauchent et plus le volume de sortie sera grand.

- **'La marge (à 0)' ou 'zeropadding'**

Parfois, il est commode de mettre des zéros à la frontière du volume d'entrée. La taille de ce 'zero-padding' est le troisième hyper paramètre. Cette marge permet de contrôler la dimension spatiale du volume de sortie. En particulier, il est parfois souhaitable de conserver la même surface que celle du volume d'entrée

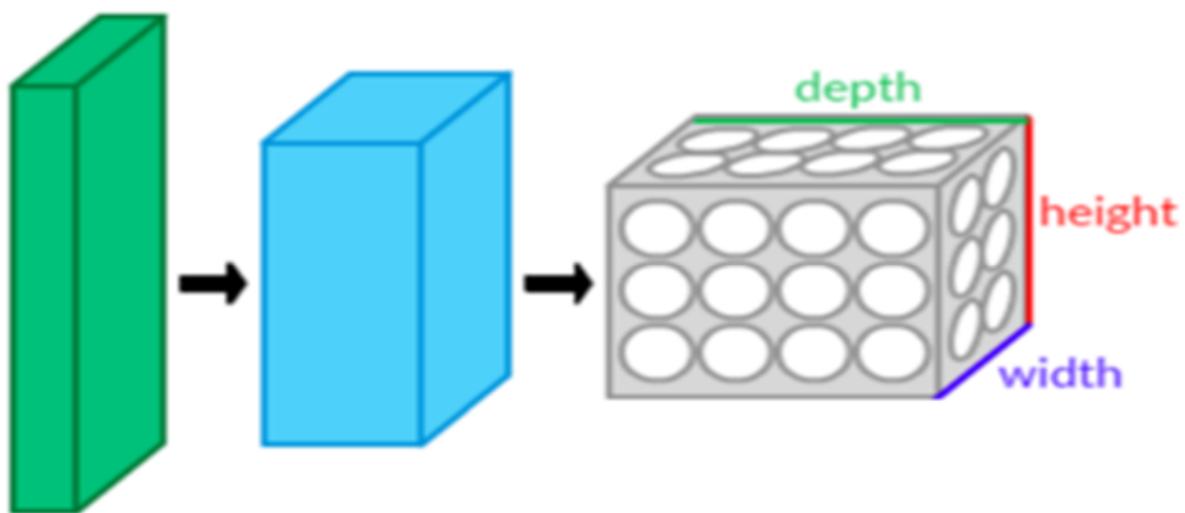


Figure 16: Une couche du CNN en 3 dimensions. (Vert = volume d'entrée, bleu = volume du champ récepteur, gris = couche de CNN, cercles = neurones artificiels indépendants).

3.3. La Couche de regroupement (POOL)

Un autre concept important des CNNs est le pooling, ce qui est une forme de sous-échantillonnage de l'image. L'image d'entrée est découpée en une série de rectangles de n pixels de côté ne se chevauchant pas (pooling). Chaque rectangle peut être vu comme une tuile. Le signal en sortie de tuile est défini en fonction des valeurs prises par les différents pixels de la tuile. Le pooling réduit la taille spatiale d'une image intermédiaire, réduisant ainsi la quantité de paramètres et de calcul dans le réseau. Il est donc fréquent d'insérer périodiquement une couche de pooling entre deux couches convolutives successives d'une architecture CNN pour contrôler sur-apprentissage (overfitting). L'opération de pooling crée aussi une forme d'invariance par translation.

La couche de pooling fonctionne indépendamment sur chaque tranche de profondeur de l'entrée et la redimensionne uniquement au niveau de la surface. La forme la plus courante est une couche de mise en commun avec des tuiles de taille 2x2 (largeur/hauteur) et comme valeur de sortie la valeur maximale en entrée. On parle dans ce cas de « Max-Pool 2x2 ».

Il est possible d'utiliser d'autres fonctions de pooling que le maximum. On peut utiliser un « average pooling » (la sortie est la moyenne des valeurs du patch d'entrée), du « L2-norm pooling ». Dans les faits, même si initialement l'average pooling était souvent utilisé il s'est avéré que le max-pooling était plus efficace car celui-ci augmente plus significativement l'importance des activations fortes. En d'autres circonstances, on pourra utiliser un pooling stochastique.

Le pooling permet de gros gains en puissance de calcul. Cependant, en raison de la réduction agressive de la taille de la représentation (et donc de la perte d'information associée), la tendance actuelle est d'utiliser de petits filtres (type 2x2). Il est aussi possible d'éviter la couche de pooling mais cela implique un risque sur-apprentissage plus important.[20]

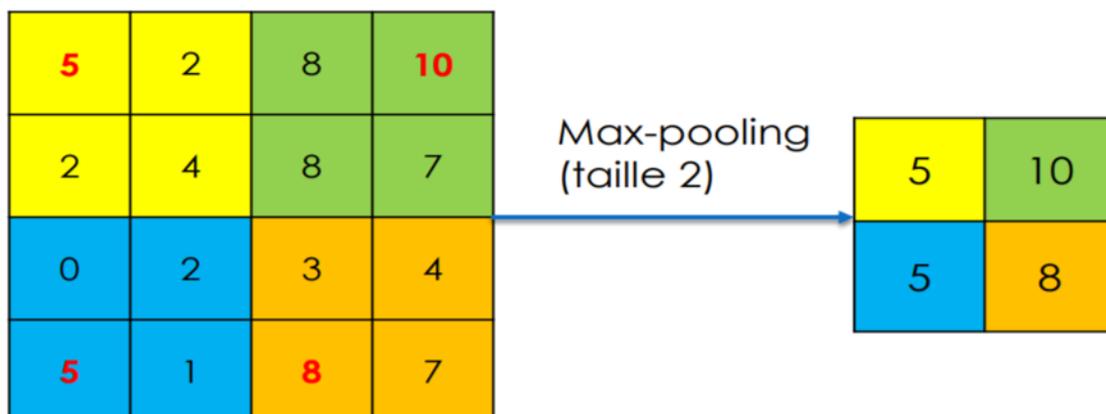


Figure 17: Pooling avec un filtre 2x2 et un pas de 2

3.4. Couche de correction [20] :

Souvent, il est possible d'améliorer l'efficacité du traitement en intercalant entre les couches de traitement une couche qui va opérer une fonction mathématique (fonction d'activation) sur les signaux de sortie. La fonction d'activation est souvent une fonction non-linéaire. Leur but est de permettre aux réseaux de neurones d'apprendre des fonctions plus complexes qu'une simple

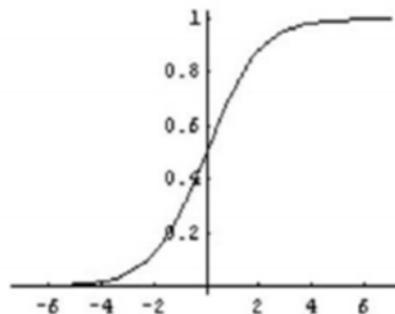
régression linéaire car le fait de multiplier les poids d'une couche cachée est juste une transformation linéaire. On a notamment :

- **Sigmoïde**

Avant l'introduction de ReLu, la plupart des réseaux neuronaux utilisent la Sigmoïde

$$g(z) = \frac{1}{1 + e^{-\lambda z}}$$

Le but premier de la fonction est de réduire la valeur d'entrée pour la réduire entre 0 et 1. En plus d'exprimer la valeur sous forme de probabilité, si la valeur en entrée est un très grand nombre positif, la fonction convertira cette valeur en une probabilité de 1. A l'inverse, si la valeur en entrée est un très grand nombre négatif, la fonction convertira cette valeur en une probabilité de 0. D'autre part, l'équation de la courbe est telle que, seules les petites valeurs influent réellement sur la variation des valeurs en sortie[6].



Ce qu'on reproche aux unités sigmoïdales c'est qu'elles saturent dans la plupart de leur domaine : elles saturent à une valeur élevée lorsque z est très positif et à une valeur faible lorsque z est très négatif et ne sont que très sensibles à leur entrée lorsque z est proche de 0. La saturation généralisée des unités sigmoïdales peut rendre l'apprentissage par la méthode de la descente de gradient très difficile. Pour cette raison, leur utilisation dans les couches cachées est maintenant déconseillée.

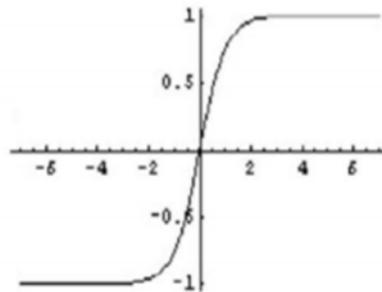
- **tangente hyperbolique.**

La fonction tangente hyperbolique est également appelée "**tanh**".

$$g(z) = \frac{1 - e^{-2z}}{1 + e^{-2z}}$$

Cette fonction ressemble à la fonction Sigmoidé. La différence avec la fonction Sigmoidé est que la fonction Tanh produit un résultat compris entre -1 et 1.

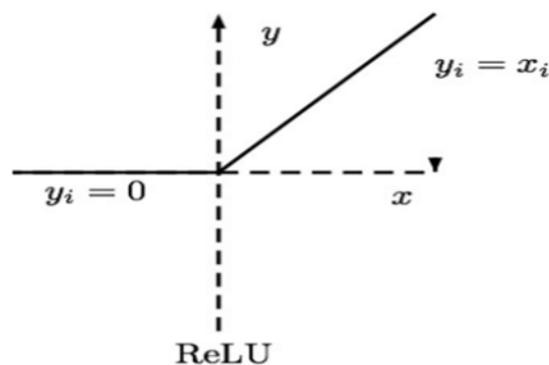
La fonction Tanh est en terme général préférable à la fonction Sigmoidé car elle est centrée sur zéro. Les grandes entrées négatives tendent vers -1 et les grandes entrées positives tendent vers 1.



Mis à part cet avantage, la fonction Tanh possède les mêmes autres inconvénients que la fonction Sigmoidé.

- **La correction ReLU (abréviation de Unités Rectifié linéaires) :**

Pour résoudre le problème de saturation des deux fonctions précédentes (Sigmoidé et Tanh) il existe la fonction ReLU (Unité de Rectification Linéaire). Cette fonction est la plus utilisée.



La fonction ReLU est interprétée par la formule :

$$f(x) = \max(0, x).$$

Si l'entrée est négative la sortie est 0 et si elle est positive alors la sortie est x. Cette fonction d'activation augmente considérablement la convergence du réseau et ne sature pas.

Mais la fonction ReLU n'est pas parfaite. Si la valeur d'entrée est négative, le neurone reste inactif, ainsi les poids ne sont pas mis à jour et le réseau n'apprend pas.

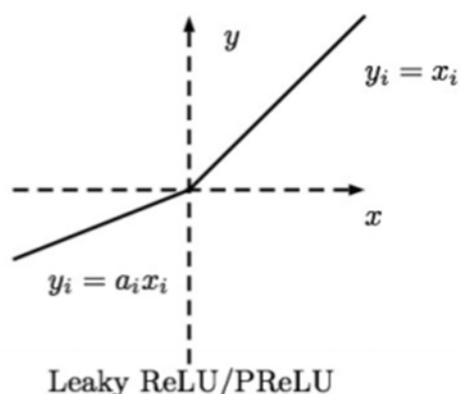
- **LeakyReLU**

La fonction LeakyReLU est interprétée par la formule:

$$f(x) = \max(0.1x, x).$$

La fonction Leaky Relu essaye de corriger la fonction ReLU lorsque l'entrée est négative. Le concept de LeakyReLU est lorsque l'entrée est négative, il aura une petite pente positive de 0,1. Cette fonction élimine quelque peu le problème d'inactivité de la fonction ReLU pour les valeurs négatives, mais les résultats obtenus avec elle ne sont pas cohérents.

Elle conserve tout de même les caractéristiques d'une fonction d'activation ReLU, c'est-à-dire efficace sur le plan des calculs, elle converge beaucoup plus rapidement et ne sature pas dans les régions positives.



La fonction d'activation par défaut recommandée pour l'utilisation des réseaux de neurones **feedforward** est **ReLU**, il en résulte la formation de réseau neuronal plusieurs fois plus rapide, sans faire une différence significative à la généralisation de précision.

Mais aussi car la version LeakyReLU est une expérimentation sans réelle plus-value.

3.5. Couche entièrement connectée (FullyconnectedLayersFC)

Après plusieurs couches de convolution et de max-pooling, le raisonnement de haut niveau dans le réseau neuronal se fait via des couches entièrement connectées. Les neurones dans une couche entièrement connectée ont des connexions vers toutes les sorties de la couche précédente. Leurs fonctions d'activations peuvent donc être calculées avec une multiplication matricielle suivie d'un décalage de polarisation.[3]

3.6. Couche de perte (LOSS)

La couche de perte spécifie comment l'entraînement du réseau pénalise l'écart entre le signal prévu et réel. Elle est normalement la dernière couche dans le réseau. Diverses fonctions de perte adaptées à différentes tâches peuvent y être utilisées. La perte « Softmax » est utilisée pour prédire une seule classe parmi K classes mutuellement exclusives. La perte par entropie croisée sigmoïde est utilisée pour prédire K valeurs de probabilité indépendante dans [0,1]. La perte euclidienne est utilisée pour régresser vers des valeurs réelles.[20]

4. Choix des hyperparamètres

Les CNNs utilisent plus d'hyperparamètres qu'un MLP standard. Même si les règles habituelles pour les taux d'apprentissage et des constantes de régularisation s'appliquent toujours, il faut prendre en considération les notions de nombre de filtres, leur forme et la forme du max pooling.

4.1.Nombre de filtres

Comme la taille des images intermédiaires diminue avec la profondeur du traitement, les couches proches de l'entrée ont tendance à avoir moins de filtres tandis que les couches plus proches de la sortie peuvent en avoir davantage. Pour égaliser le calcul à chaque couche, le produit du nombre de caractéristiques et le nombre de pixels traités est généralement choisi pour être à peu près constant à travers les couches. Pour préserver l'information en entrée, il faudrait maintenir le nombre de sorties intermédiaires (nombre d'images intermédiaire multiplié par le nombre de positions de pixel) pour être croissante (au sens large) d'une couche à l'autre. Le nombre d'images intermédiaires contrôle directement la puissance du système, dépend du nombre d'exemples disponibles et la complexité du traitement. [5]

4.2. Forme de filtres

Les formes de filtre varient grandement dans la littérature. Ils sont généralement choisis en fonction de l'ensemble de données. Les meilleurs résultats sur les images de MNIST (28x28) sont habituellement dans la gamme de 5x5 sur la première couche, tandis que les ensembles de données d'images naturelles (souvent avec des centaines de pixels dans chaque dimension) ont tendance à utiliser de plus grands filtres de première couche de 12x12, voire 15x15.

Le défi est donc de trouver le bon niveau de granularité de manière à créer des abstractions à l'échelle appropriée et adaptée à chaque cas. [3]

4.3. Forme de pooling

Les valeurs typiques sont 2*2 ou 3*3. De très grands volumes d'entrée peuvent justifier un pooling 4x4 dans les premières couches. Cependant, le choix de formes plus grandes va considérablement réduire la dimension du signal, et peut entraîner la perte de trop d'information.[20]

C'est une opération simple qui consiste à remplacer un carré de pixels (généralement 2x2 ou 3x3) par une valeur unique. De cette manière, l'image diminue en taille et se retrouve simplifiée (lissée)

Il existe plusieurs types de pooling :

- Le « *max pooling* », qui revient à prendre la valeur maximale de la sélection. C'est le type le plus utilisé car il est rapide à calculer (immédiat), et permet de simplifier efficacement l'image
- Le « *meanpooling* » (ou *averagepooling*), soit la moyenne des pixels de la sélection : on calcule la somme de toutes les valeurs et on divise par le nombre de valeurs. On obtient ainsi une valeur intermédiaire pour représenter ce lot de pixels
- Le « *sumpooling* », c'est la moyenne sans avoir divisé par le nombre de valeurs (on ne calcule que leur somme) [7]

5. Différence entre réseaux de neurones convolutifs comparés et perceptron multicouche

Bien qu'efficaces pour le traitement d'images, les perceptrons multicouches (MLP) ont des difficultés à gérer des images de grande taille, en raison de la croissance exponentielle du nombre de connexions avec la taille de l'image, du fait que chaque neurone est « totalement connecté » à chacun des neurones de la couche précédente et suivante. Les réseaux de neurones convolutifs, dont le principe est inspiré de celui du cortex visuel des vertébrés,

limite au contraire le nombre de connexions entre un neurone et les neurones des couches adjacentes, ce qui diminue drastiquement le nombre de paramètres à apprendre. Pour un réseau profond tel que AlexNet par exemple, plus de 90 % des paramètres à apprendre sont dus aux 3 couches « complètement connectées » les plus profondes, et le reste concerne les (5) couches convolutives.

Exemple, si on prend une image de taille $32 \times 32 \times 3$ (32 de large, 32 de haut, 3 canaux de couleur), un seul neurone entièrement connecté dans la première couche cachée du MLP aurait 3 072 entrées ($32 \times 32 \times 3$). Une image 200×200 conduirait ainsi à traiter 120 000 entrées par neurone ce qui, multiplié par le nombre de neurones, devient énorme.

Alors que les réseaux de neurones convolutifs visent à limiter le nombre d'entrées tout en conservant la forte corrélation « spatialement locale » des images naturelles. Par opposition aux MLP, les CNN ont les traits distinctifs suivants

1. **Connectivité locale** : grâce au champ récepteur qui limite le nombre d'entrées du neurone, tout en conservant l'architecture MLP, les réseaux de neurones convolutifs assurent ainsi que les « filtres » produisent la réponse la plus forte à un motif d'entrée spatialement localisé, ce qui conduit à une représentation parcimonieuse de l'entrée. Une telle représentation occupe moins d'espace en mémoire. De plus, le nombre de paramètres à estimer étant réduit, leur estimation (statistique) est plus robuste pour un volume de données fixé (comparé à un MLP).
2. **Poids partagés** : dans les réseaux de neurones convolutifs, les paramètres de filtrage d'un neurone (pour un champ récepteur donné) sont identiques pour tous les autres neurones d'un même noyau (traitant tous les autres champs récepteurs de l'image). Ce paramétrage (vecteur de poids et biais) est défini dans une « carte de fonction ».
3. **Invariance à la translation** : comme tous les neurones d'un même noyau (filtre) sont identiques, le motif détecté par ce noyau est indépendant de localisation spatiale dans l'image

Toutes ces propriétés permettent aux réseaux de neurones à convolution d'obtenir une meilleure robustesse dans l'estimation des paramètres sur des problèmes d'apprentissage puisque, pour une taille de corpus d'apprentissage fixée, la quantité de données par paramètres est plus grande. Le partage de poids permet aussi de réduire considérablement le nombre de paramètres libres à apprendre, et ainsi les besoins en mémoire pour le fonctionnement du

réseau. La diminution de l'empreinte mémoire permet l'apprentissage de réseaux plus grands donc souvent plus puissants.

5.1. Avantages de CNNs

Un avantage majeur des réseaux convolutifs est l'utilisation d'un poids unique associé aux signaux entrant dans tous les neurones d'un même noyau de convolution. Cette méthode réduit l'empreinte mémoire, améliore les performances et permet une invariance du traitement par translation. C'est le principal avantage du CNN par rapport au MLP, qui lui considère chaque neurone indépendant et donc affecte un poids différent à chaque signal entrant. Lorsque le volume d'entrée varie dans le temps (vidéo ou son), il devient intéressant de rajouter un paramètre de temporisation (delay) dans le paramétrage des neurones. On parlera dans ce cas de réseau neuronal à retard temporel (TDNN)

Comparés à d'autres algorithmes de classification de l'image, les réseaux de neurones convolutifs utilisent relativement peu de pré-traitement. Cela signifie que le réseau est responsable de faire évoluer tout seul ses propres filtres (apprentissage sans supervision), ce qui n'est pas le cas d'autres algorithmes plus traditionnels. L'absence de paramétrage initial et d'intervention humaine est un atout majeur des CNN.[18]

6. Approches & Méthodes de Classification d'Images

6.1. Définition de la classification

Classifier une image est une tâche ou une série de méthodes qu'une théorie unifiée pour pouvoir utiliser les images pour les analyses complémentaires ou pour la cartographie. Elle consiste à déterminer des procédures permettant de rassembler l'ensemble des pixels de l'images en un nombre limité de classes correspondant aux grands éléments structuraux de l'image. [22]

La qualité de la classification peut être jugée sur la base des deux critères suivants :

- Les classes générées doivent être les plus différents possibles les unes des autres vis-à-vis de certaines caractéristiques
- Chaque classe doit être la plus homogène possible vis-à-vis de ces caractéristiques.

6.2. Les motivations de la Classification des images :

La classification des images consiste à répartir systématiquement des images selon des classes établies au préalable, classer une image lui fait correspondre une classe, marquant ainsi sa parenté avec d'autres images. En général reconnaître une image est une tâche aisée pour un humain au fil de son existence, il a acquis des connaissances qui lui permettent de s'adapter aux variations qui résultent de conditions différentes d'acquisition. Il lui est par exemple relativement simple de reconnaître un objet dans plusieurs orientations partiellement caché par un autre de près ou de loin et selon diverses illuminations. Toutefois les progrès technologiques en termes d'acquisition d'images (microscopes, caméras, capteurs) et de stockage engendrent des bases de données riches en information et multiplient les domaines d'applications, il devient alors difficile pour l'humain d'analyser le nombre important d'images, le temps requis le caractère répétitif de la tâche et la concentration nécessaire sont problématiques. Toutefois celle-ci n'est pas forcément aisée pour un programme informatique pour lequel une image est un ensemble de valeur numérique. L'objectif de la classification d'images est d'élaborer un système capable d'affecter une classe automatiquement à une image. Ainsi, ce système permet d'effectuer une tâche d'expertise qui peut s'avérer coûteuse à acquérir pour un être humain en raison notamment de contraintes physiques comme la concentration, la fatigue ou le temps nécessaire par un volume important de données images. Les applications de la classification automatique d'images sont nombreuses et vont de l'analyse de documents à la médecine en passant par le domaine militaire. Ainsi on retrouve des applications dans le domaine médical comme la reconnaissance de cellules et de tumeurs, la reconnaissance d'écriture manuscrite pour les chèques les codes postaux. Dans le domaine urbain comme la reconnaissance de panneaux de signalisation la reconnaissance de piétons la détection de véhicules la reconnaissance de bâtiments pour aider à la localisation.

Dans le domaine de la biométrie comme la reconnaissance de visage, d'empreintes, d'iris. Le point commun à toutes ces applications est qu'elles nécessitent la mise en place d'une chaîne de traitement à partir des images disponibles composée de plusieurs

étapes afin de fournir en sortie une décision. Chaque étape de la mise en place d'un tel système de classification nécessite la recherche de méthodes appropriées pour une performance globale optimale à savoir la phase d'extraction de caractéristiques et la phase d'apprentissage. Typiquement, nous disposons de données images desquelles il nous faut extraire des informations pertinentes traduites sous formes de vecteurs numériques. Cette phase d'extraction nous permet de travailler dans un espace numérique. Il s'agit ensuite d'élaborer dans la phase d'apprentissage, à partir de ces données initiales, une fonction de décision pour décider de l'appartenance d'une donnée nouvelle à l'une des classes en présence.[3]

6.3. Différentes méthodes de la classification et l'apprentissage :

De nombreuses méthodes classiques ont été consacrées, elles peuvent être séparées en deux grandes catégories : les méthodes de classification supervisée et les méthodes de classification non supervisée.

6.3.1. Méthodes supervisées :

Dans ce cas, on dispose d'un ensemble de données étiquetées, ou d'exemples qui se sont vus associés une classe par un professeur ou un expert. Cet ensemble d'exemples constitue la base d'apprentissage.

Les méthodes d'apprentissage supervisé se donnent alors comme objectif général de construire à partir de la base d'apprentissage une fonction de classement. Une telle fonction permet, à partir de la description d'un objet, de reconnaître un attribut particulier.

C'est-à-dire le système va devoir comparer entre les caractéristiques qui constituent chaque classe pour ensuite les comparer aux données non classifiées pour déduire à quelle classe elles appartiennent.[4]

6.3.2. Méthodes non supervisées :

L'apprentissage non-supervisé, encore appelé apprentissage à partir d'observations ou découverte, consiste à déterminer une classification « sensée » à partir d'un ensemble d'objets ou de situations données (des exemples non étiquetés). [5]

On dispose d'une masse de données indifférenciées, et l'on désire savoir si elles possèdent une quelconque structure de groupes. Il s'agit d'identifier une éventuelle tendance des données à être regroupées en classes. Ce type d'apprentissage, encore appelé Clustering ou Cluster

Analysais, se trouve en classification automatique et en taxinomie numérique. Cette forme de classification existe depuis des temps immémoriaux. Elle concerne notamment les sciences de la nature, les classifications des documents et des livres mais également la classification des sciences élaborées au cours des siècles par les philosophes.[27]

Apprentissage supervisé	Apprentissage non supervisé
<ul style="list-style-type: none">- Nombre de classe connues- Sur entraînement- Utilisé pour classifier des données futures	<ul style="list-style-type: none">- Nombre de classes inconnues- Pas de connaissance préalable- Utilisé pour comprendre et explorer les données

Tableau 1 : différence entre apprentissage supervisé et non supervisé.

Donc La différence distinct entre l'apprentissage supervisé à l'apprentissage non-supervisé est le fait que l'apprentissage non-supervisé cherche à trouver des partitions de modèles par lui-même, l'extraction des données est donc descriptive.

L'apprentissage supervisé est utilisé quand l'utilisateur sait labelliser les informations, l'extraction des données est donc prédictive.

6.4. Indicateurs de performance en classification

6.4.1. Matrice de confusion [3] :

Prenons l'exemple d'un classifieur binaire, c'est-à-dire, qui prédit 2 classes notées classe 0 et classe 1. Pour mesurer les performances de ce classifieur, il est d'usage de distinguer 4 types d'éléments classés pour la classe voulue :

- Vrai positif VP : Élément de la classe 1 correctement prédit
- Vrai négatif VN : Élément de la classe 0 correctement prédit
- Faux positif FP : Élément de la classe 1 mal prédit
- Faux négatif FN : Élément de la classe 0 mal prédit

Ces informations peuvent être rassemblés et visualisés sous forme de tableau dans une matrice de confusion. Dans le cas d'un classifieur binaire, on obtient :

		Classe prédite	
		Classe 0	Classe 1
Classe réelle	Classe 0	VN	FN
	Classe 1	FP	VP

En particulier, si la matrice de confusion est diagonale, le classifieur est parfait. Notons que la matrice de confusion est aussi généralisable lorsqu'il y a $k > 2$ classes à prédire.

Il est possible de calculer plusieurs indicateurs résumant la matrice de confusion. Par exemple si nous souhaitons rendre compte de la qualité de la prédiction sur la classe 1, on définit :

- **Précision** : Proportion d'éléments bien classés pour une classe donnée :

$$Precision_{de\ la\ classe\ 1} = \frac{VP}{VP + FP}$$

- **Rappel (sensibilité)**: Proportion d'éléments bien classés par rapport au nombre d'éléments de la classe à prédire :

$$Rappel_{de\ la\ classe\ 1} = \frac{VP}{VP + FN}$$

- **F-mesure** : Mesure de compromis entre précision et rappel :

$$F\text{-mesure}_{de\ la\ classe\ 1} = \frac{2 * (Précision + Rappel)}{Précision + Rappel}$$

Il est possible de calculer tous ces indicateurs pour chaque classe. La moyenne sur chaque classe de ces indicateurs donne des indicateurs globaux sur la qualité du classifieur

$$Precision = \frac{1}{k} \sum_{i=1}^k \frac{VP_i}{VP_i + FP_i}$$

$$Rappel = \frac{1}{k} \sum_{i=1}^k \frac{VP_i}{VP_i + FN_i}$$

$$F\text{-mesure} = \frac{2 * (Précision + Rappel)}{Précision + Rappel}$$

7. Classification des images et l'apprentissage machine :

Les méthodes manuelles se sont avérées très difficiles à appliquer pour des tâches en apparence très simples comme la classification des images, la reconnaissance d'objets dans les images ou la reconnaissance vocale. Les données venant du monde réel les échantillons d'un son ou les pixels d'une image sont complexes, variables et entachées de bruit. Pour une machine, une image est un tableau de nombres indiquant la luminosité (ou la couleur) de chaque pixel, et un signal sonore une suite de nombres indiquant la pression de l'air à chaque instant. Comment une machine peut-elle transcrire la suite de nombres d'un signal sonore en série de mots tout en ignorant le bruit ambiant, l'accent du locuteur et les particularités de sa voix ? Comment une machine peut-elle identifier un chien ou une chaise dans le tableau de nombres d'une image quand l'apparence d'un chien ou d'une chaise et des objets qui les entourent peut varier infiniment ? Il est virtuellement impossible d'écrire un programme qui fonctionnera de manière robuste dans toutes les situations. C'est là qu'intervient l'apprentissage machine (que l'on appelle aussi apprentissage automatique). C'est l'apprentissage qui anime les systèmes de toutes les grandes entreprises d'Internet. Elles l'utilisent depuis longtemps pour filtrer les contenus indésirables, ordonner des réponses à une recherche, faire des recommandations, ou sélectionner les informations intéressantes pour chaque utilisateur.

Un système entraînable peut être vu comme une boîte noire avec une entrée, par exemple une image, un son, ou un texte, et une sortie qui peut représenter la catégorie de l'objet dans l'image, le mot prononcé, ou le sujet dont parle le texte. On parle alors de systèmes de classification ou de reconnaissance des formes. Dans sa forme la plus utilisée, l'apprentissage machine est supervisé : on montre en entrée de la machine une photo d'un objet, par exemple une voiture, et on lui donne la sortie désirée pour une voiture. Puis on lui montre la photo d'un chien avec la sortie désirée pour un chien. Après chaque exemple, la machine ajuste ses paramètres internes de manière à rapprocher sa sortie de la sortie désirée. Après avoir montré à la machine des milliers ou des millions d'exemples étiquetés avec leur catégorie, la machine devient capable de classer correctement la plupart d'entre eux. Mais ce qui est plus intéressant, c'est qu'elle peut aussi classer correctement des images de voiture ou de chien

qu'elle n'a jamais vues durant la phase l'apprentissage. C'est ce qu'on appelle la capacité de généralisation. Jusqu'à récemment, les systèmes de reconnaissance des images classiques étaient composés de deux blocs : un extracteur de caractéristiques (featureextractor en anglais), suivi d'un classifieur entraînable simple. L'extracteur de caractéristiques est programmé « à la main », et transforme le tableau de nombres représentant l'image en une série de nombres, un vecteur de caractéristiques, dont chacun indique la présence ou l'absence d'un motif simple dans l'image. Ce vecteur est envoyé au classifieur, dont un type commun est le classifieur linéaire. Ce dernier calcule une somme pondérée des caractéristiques : chaque nombre est multiplié par un poids (positif ou négatif) avant d'être sommé. Si la somme est supérieure à un seuil, la classe est reconnue. Les poids forment une sorte de « prototype » pour la classe à laquelle le vecteur de caractéristiques est comparé. Les poids sont différents pour les classifieurs de chaque catégorie, et ce sont eux qui sont modifiés lors de l'apprentissage. Les premières méthodes de classification linéaire entraînable datent de la fin des années cinquante et sont toujours largement utilisées aujourd'hui. Elles prennent les doux noms de perceptron ou régression logistique. [3]

8. Classification des images et les réseaux de neurones

Le problème de l'approche classique de la reconnaissance des images est qu'un bon extracteur de caractéristiques est très difficile à construire, et qu'il doit être repensé pour chaque nouvelle application. C'est là qu'intervient l'apprentissage profond ou deep learning en anglais. C'est une classe de méthodes dont les principes sont connus depuis la fin des années 1980, mais dont l'utilisation ne s'est vraiment généralisée que depuis 2012, environ. L'idée est très simple : le système entraînable est constitué d'une série de modules, chacun représentant une étape de traitement. Chaque module est entraînable, comportant des paramètres ajustables similaires aux poids des classifieurs linéaires. Le système est entraîné de bout en bout : à chaque exemple, tous les paramètres de tous les modules sont ajustés de manière à rapprocher la sortie produite par le système de la sortie désirée. Le qualificatif profond vient de l'arrangement de ces modules en couches successives.

Pour pouvoir entraîner le système de cette manière, il faut savoir dans quelle direction et de combien ajuster chaque paramètre de chaque module. Pour cela il faut calculer un gradient, c'est-à-dire pour chaque paramètre ajustable, la quantité par laquelle l'erreur en sortie augmentera ou diminuera lorsqu'on modifiera le paramètre d'une quantité donnée. Le calcul de ce gradient se fait par la méthode de rétropropagation, pratiquée depuis le milieu des années 1980. Dans sa réalisation la plus commune, une architecture profonde peut être vue comme un réseau multicouche d'éléments simples, similaires aux classifieurs linéaires, interconnectés par des poids entraînaibles. C'est ce qu'on appelle un réseau neuronal multicouche.

Pourquoi neuronal ? Un modèle extrêmement simplifié des neurones du cerveau les voit comme calculant une somme pondérée et activant leur sortie lorsque celle-ci dépasse un seuil. L'apprentissage modifie les efficacités des synapses, les poids des connexions entre neurones. Un réseau neuronal n'est pas un modèle précis des circuits du cerveau, mais est plutôt vu comme un modèle conceptuel ou fonctionnel. Le réseau neuronal est inspiré du cerveau un peu comme l'avion est inspiré de l'oiseau. Ce qui fait l'avantage des architectures profondes, c'est leur capacité d'apprendre à représenter le monde de manière hiérarchique. Comme toutes les couches sont entraînaibles, nul besoin de construire un extracteur de caractéristiques à la main. L'entraînement s'en chargera.

De plus, les premières couches extrairont des caractéristiques simples (présence de contours) que les couches suivantes combineront pour former des concepts de plus en plus complexes et abstraits : assemblages de contours en motifs, de motifs en parties d'objets, de parties d'objets en objets, etc.

9. Deep Learning et agriculture

Les réseaux de neurones (profonds ou non) pour des cas agricoles sont explorés/utilisés de façon importante par les chercheurs. Le graphique ci-contre, issu de l'étude montre la part des réseaux de neurones par rapport aux autres algorithmes utilisés pour des cas agricoles. Pour une grande part, il s'agit de travaux basés sur la vision : l'analyse d'images. Deux familles de réseaux sont largement utilisées : les réseaux convolutifs, dans une très large majorité, et les réseaux récurrents, en augmentation. Les Réseaux de type GAN, plus récents, ne sont utilisés que marginalement, pour débruiter des images par exemple

Notons enfin également que les articles scientifiques peuvent évoquer des usages agricoles au titre de simples cadres d'utilisation (transposables à d'autres domaines) pour l'optimisation d'architectures de réseaux ou de méthodes (par exemple utilisation d'autres algorithmes pour pré-entraîner les réseaux de façon non supervisée

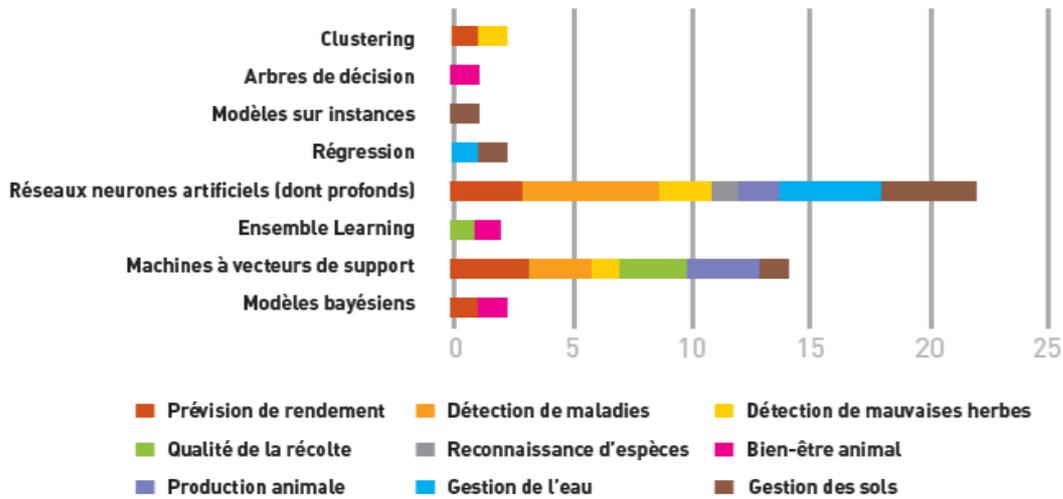


Figure 18 : Nombre de modèles de machine Learning pour des usages agricoles d'après étude couvrant une période de 2004 à nos jours.

Conclusion :

Nous avons consacré ce chapitre à la présentation des approches et méthodes de la classification ainsi que leurs intérêts dans le domaine d'imagerie et on a parlé aussi sur l'utilisation des réseaux de neurones dans ce domaine.

Nous avons aussi présenté les notions importantes qui sont en relation avec l'apprentissage profond (Définition, architecture, topologies et fonctionnement). Aussi on a donné une vision générale sur les CNNs.

Dans le prochain chapitre, nous présenterons notre approche utilisée pour traiter des images captées par un drone et les classifiées en fonction des informations données.

Chapitre 3 :
Implémentation d'une application de
classification

Introduction :

Dans les chapitres précédents on a vu les notions importantes de l'apprentissage profond (Définition, architecture, topologies et fonctionnement) , Aussi on a introduit les réseaux de neurones leurs concepts et différentes notions ainsi on a présenté les différentes approches et méthodes de la classification d'images en utilisant les réseaux de neurones .

Dans ce chapitre, nous allons présenter la conception de notre application qui a pour but principal la classification d'images agricoles en utilisant le Deep Learning et précisément réseaux de neurones convolutifs .

Les réseaux de neurones convolutifs sont l'outil de choix pour la classification d'images . Ce sont des algorithmes du Deep Learning, objets d'intenses recherches... dont la richesse peut impressionner, En pratique, très peu de personnes forment entièrement un réseau de convolution (avec une initialisation aléatoire), car il est relativement rare de disposer d'un jeu de données de taille suffisante, Mais il est possible de créer très simplement des modèles performants : avec peu d'images, peu de capacités de calcul et sans maîtrise des arcanes algorithmiques. Toute l'astuce consiste à adapter des réseaux pré-entraînés disponible publiquement sur d'autres problèmes.

On exploite la connaissance acquise sur un problème de classification générale pour l'appliquer de nouveau à un problème particulier.

On a implémenté dans ce travail deux principales architectures des réseaux de neurones convolutifs (deux réseaux pré-entraînés) Alexnet, Resnet 50 et pour cela on a utilisé le langage MATLAB (version 2018) .

On a appliqué chacun de ses réseaux pré-entraînés sur deux bases d'images dataset 1 qui est d'un ensemble d'images agricoles de différentes plantes et qui est divisé en deux classes selon la densité de mauvaises herbes présente dans chaque image (Classe A constituée d'images qui ont une couche dense de mauvaises herbes et Classe B constituée d'images qui ont un taux très réduit de mauvaises herbes) et le dataset 2 contient 12 classes d'images agricoles (Ail ,Aubergine, Betterave ,Haricot Rouge ,Haricot vert ,Herbe noire ,Lentille ,Mais ,Moutarde des champs ,Poivrier noir , Salade , Tomate).

1. Logiciel utilisé dans l'implémentation :

On a utilisé le langage Matlab pour mettre en œuvre notre application.

MATLAB_ (Matrix LABORatory) est un logiciel interactif basé sur le calcul matriciel (permet de manipuler des matrices) . Il est utilisé dans les calculs scientifiques et les problèmes d'ingénierie parce qu'il permet de résoudre des problèmes numériques complexes en moins de temps requis par les langages de programmation courant, et ce grâce à une multitude de fonctions intégrées et à plusieurs programmes outils testés et regroupés selon usage dans des dossiers appelés boîtes à outils ou "toolbox".

Son objectif, par rapport aux autres langages, est de simplifier au maximum la transcription en langage informatique d'un problème mathématique, en utilisant une écriture la plus proche possible du langage naturel scientifique.[18]

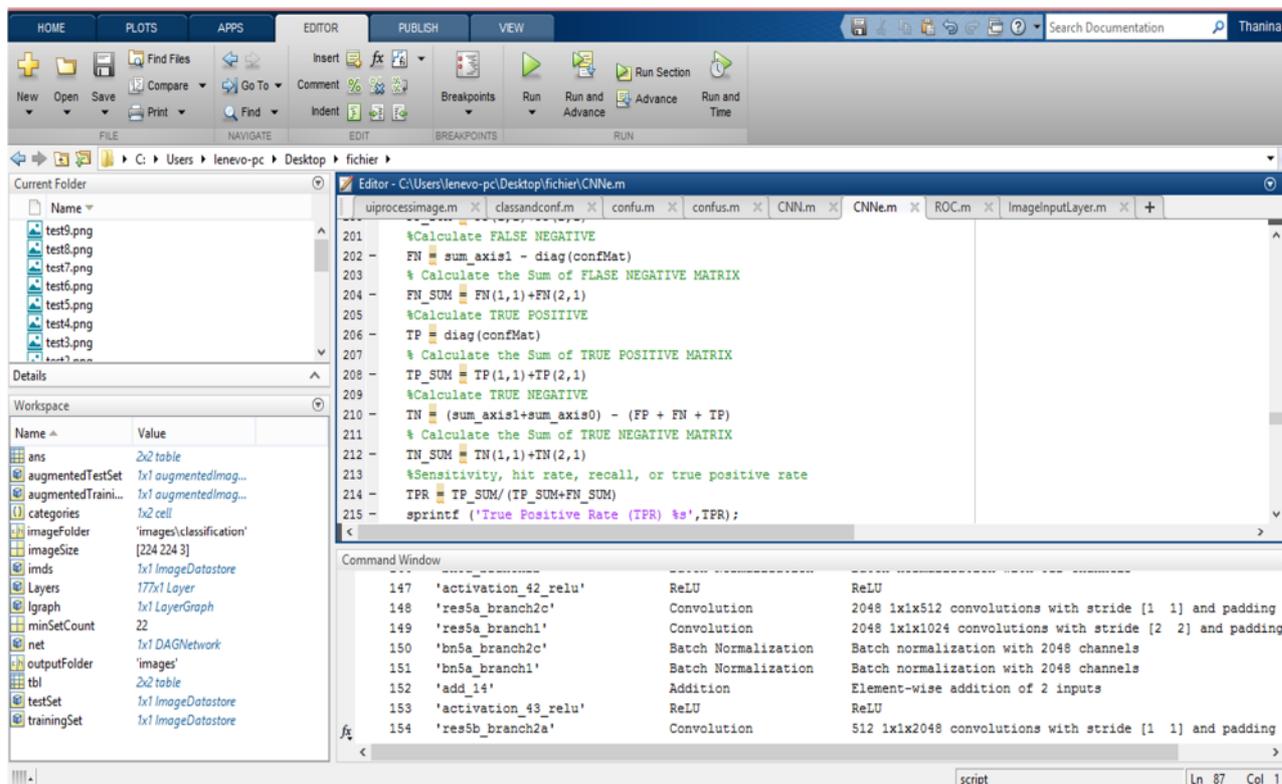


Figure 19 : Environnement de Matlab

2. Configuration matérielle utilisé dans l'implémentation :

On a utilisé dans notre travail :

- Un PC portable Dell i7 CPU 2.60 GHZ.
- RAM de taille 8 GO.
- Disque dur de taille 250 SSD.
- Système d'exploitation Windows 10 (64 bits).
- Carte graphique Intel® HD Graphics 5500

3. Qu'est-ce qu'un réseau pré-entraîné ?

Créer un nouveau réseau de neurones convolutif est coûteux en termes d'expertise, de matériel et de quantité de données annotées nécessaires.

Il s'agit d'abord de fixer l'architecture du réseau, c'est-à-dire le nombre de couches, leurs tailles et les opérations matricielles qui les connectent. L'entraînement consiste alors à optimiser les coefficients du réseau pour minimiser l'erreur de classification en sortie. Cet entraînement peut prendre plusieurs mois voire des années pour les meilleurs CNN.

Pour des usages pratiques, il est possible d'exploiter la puissance des CNN sans être un expert du domaine, avec du matériel accessible et une quantité raisonnable de données annotées. Toute la complexité de création de CNN peut être évitée en adaptant des réseaux pré-entraînés disponibles publiquement.

Un réseau pré-entraîné pour classification d'images est un réseau qui a déjà appris à extraire des caractéristiques puissantes et informatives d'images et l'utiliser comme point de départ pour apprendre une nouvelle tâche.

Ces réseaux pré-entraînés peuvent classer les images en 1 000 catégories d'objets, telles que le clavier, le café, le crayon et de nombreux animaux, les herbes[23]

On les utilise pour différentes tâches :

- **La classification** : On applique directement des réseaux pré-entraînés aux problèmes de classification. Pour classer une nouvelle image, on utilise la fonction (Classify).
- **Extraction des caractéristiques** : L'extraction automatique de caractéristiques exploite uniquement la partie convolutive d'un réseau pré-entraîné. Elle l'utilise comme extracteur de caractéristiques des images, pour alimenter le classifieur de votre choix.

- **Apprentissage par transfert** : on prend des couches d'un réseau formé sur un grand ensemble de données et ajustez un nouveau jeu de données.

Le graphique ci-dessous compare la précision de validation avec le temps nécessaire pour effectuer une prédiction à l'aide du réseau. Un bon réseau a une grande précision et est rapide. Le graphique affiche la précision de la classification en fonction du temps de prédiction avec un GPU moderne (un NVIDIA[®] TITAN Xp). Le temps de prédiction est mesuré par rapport au réseau le plus rapide. La surface de chaque marqueur est proportionnelle à la taille du réseau sur le disque

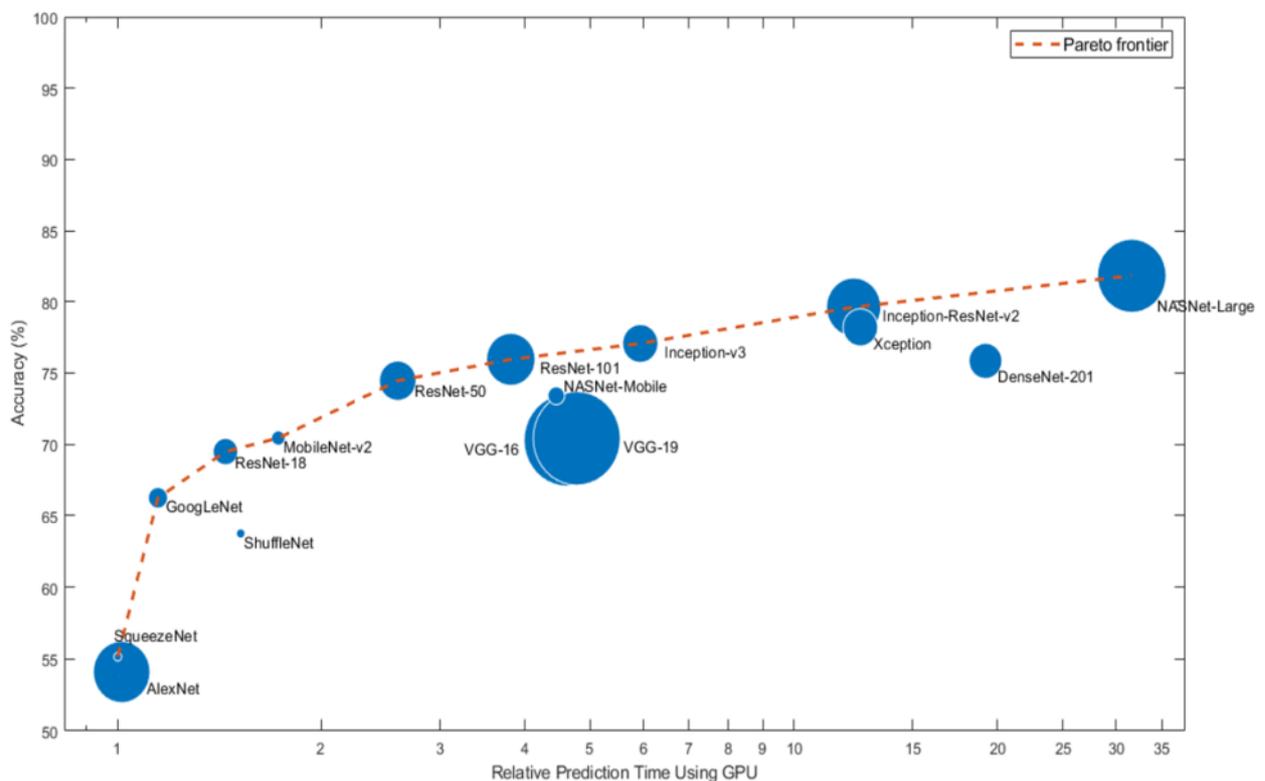


Figure 20: Le graphique précision de la classification en fonction du temps de prédiction avec un GPU.

4. Les réseaux de neurones pré-entraînés :

4.1. Alexnet :

AlexNet est un réseau de neurones de convolution pré-entraîné formé sur plus d'un million d'images de la base de données ImageNet . Le réseau a une profondeur de 8 couches et permet de classer les images en 1 000 différentes catégories d'objets. En conséquence, le réseau a appris des représentations de fonctions riches pour une large gamme d'images. Le réseau a une taille d'entrée d'image de 227 sur 227[24]

4.2. Resnet-50 :

le réseau dit de neurones résiduels (ResNet), ResNet-50 est un modèle pré-entraîné ayant été formé sur un sous-ensemble d'une base de données ImageNet .

Le modèle est formé sur plus d'un million d'images, il compte 50 couches et pouvant classer les images en 1000 catégories d'objets (par exemple, clavier, souris, crayon ,herbes). [24]

5. Architecture de notre réseau :

Au cours de nos expérimentations, nous avons utilisé deux modèles (modèle 1, modèle 2) avec différents architectures, où et on a utilisé nos deux bases de données (dataset1 et dataset 2) ,et on a utilisé une base de données test pour pouvoir mesurer le taux d'erreurs et le taux de précision de chaque modèle .

Dans ce qui suit on présente l'architecture des deux modèles

5.1. Architecture du modèle 01 (AlexNet) :

AlexNet comprend 5 couches convolutives et 3 couches entièrement connectées (FC). Les noyaux de convolution multiples (filtres) extraient des caractéristiques intéressantes dans une image. Dans une seule couche de convolution, il y a généralement plusieurs noyaux de la même taille. Par exemple, la première couche de convergence d'AlexNet contient 96 noyaux de taille 11x11x3. Notez que la largeur et la hauteur du noyau sont généralement les mêmes et que la profondeur correspond au nombre de canaux.

Les deux premières couches convolutives sont suivies des couches superposées de regroupement maximal. Les troisième, quatrième et cinquième couches convolutives sont connectées directement. La cinquième couche de convolution est suivie par une couche de regroupement maximum se chevauchant, dont la sortie se trouve dans une série de deux couches entièrement connectées. La deuxième couche entièrement connectée alimente un classifieur softmax avec 1000 étiquettes de classe. La non-linéarité ReLU est appliquée après toutes les couches de convolution et entièrement connectées. La non-linéarité ReLU des première et seconde couche de convolution est suivie d'une étape de normalisation locale avant la mise en pool.

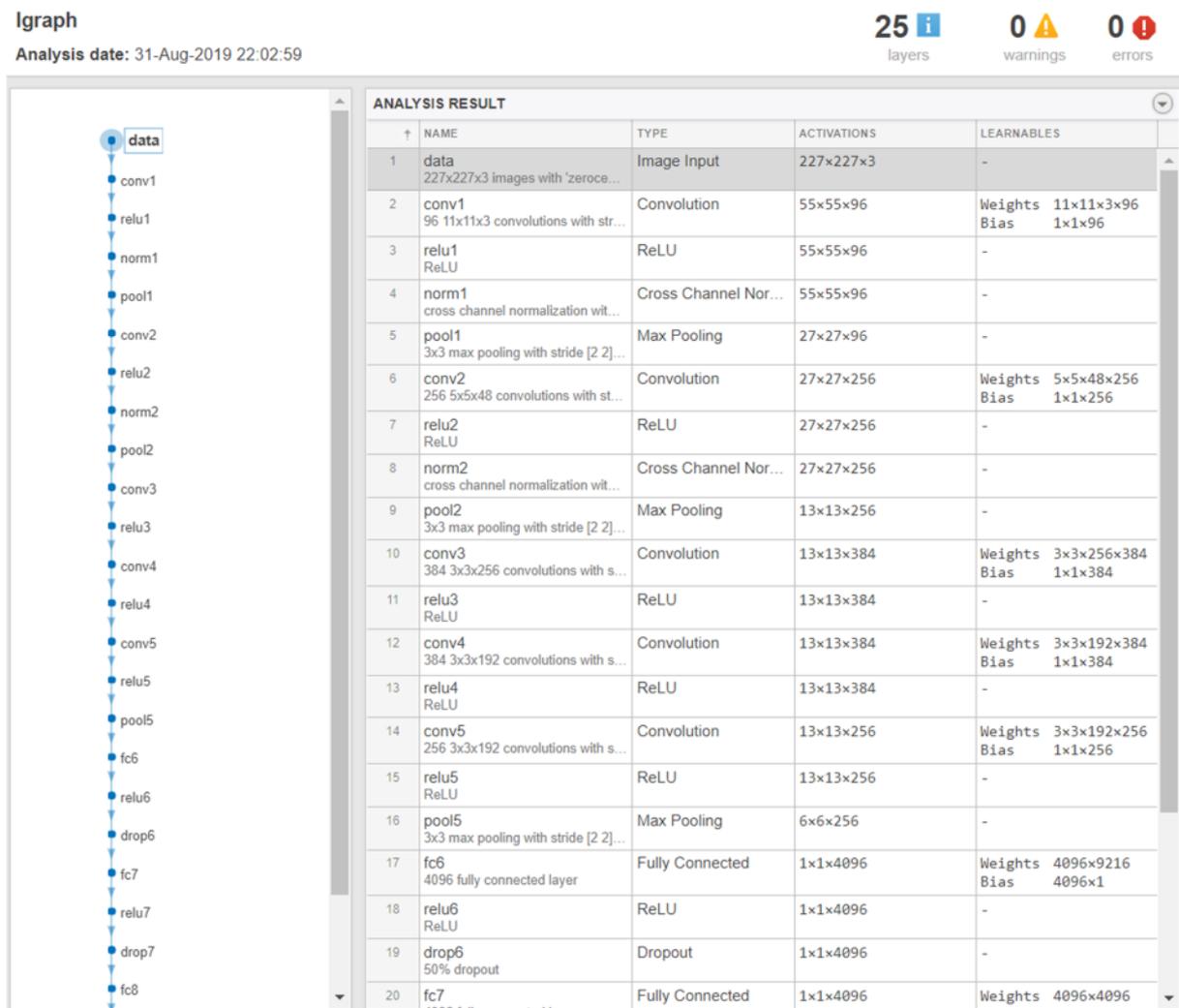


Figure 21 : Architecture de AlexNet sous Matlab

5.2. Architecture du modèle 02 (Resnet-50) :

ResNet 50 gagne facilement en précision grâce à une profondeur considérablement accrue, produisant des résultats meilleurs que ceux des réseaux précédents.

Dans l'apprentissage résiduel, au lieu d'essayer d'apprendre certaines caractéristiques, nous essayons d'apprendre quelque chose de résiduel. Le résidu peut être simplement compris comme une soustraction d'une caractéristique apprise à partir de l'entrée de cette couche. ResNet le fait en utilisant des connexions raccourcies (reliant directement l'entrée de la nième couche à une $(n + x)$ ème couche. Il est prouvé que la formation de cette forme de réseau est plus facile que la formation de simples réseaux neuronaux de convolution profonde et que le problème de la précision dégradante est résolu, c'est le concept fondamental de ResNet.

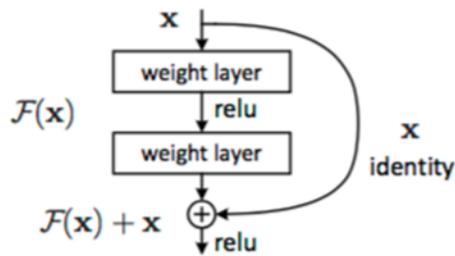


Figure 22 : bloc résiduel

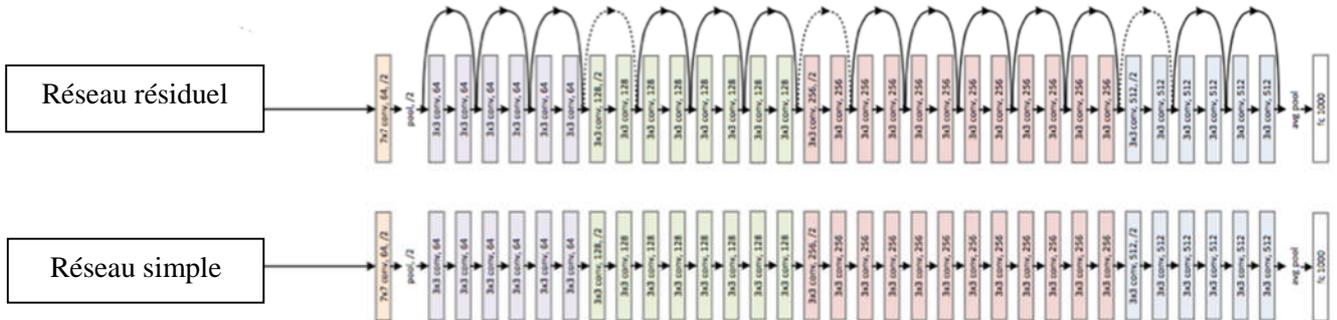


Figure 23: schéma d'un réseau résiduel et un réseau simple

Voici un schéma représentatif de ResNet50 visualisé grâce à Matlab :

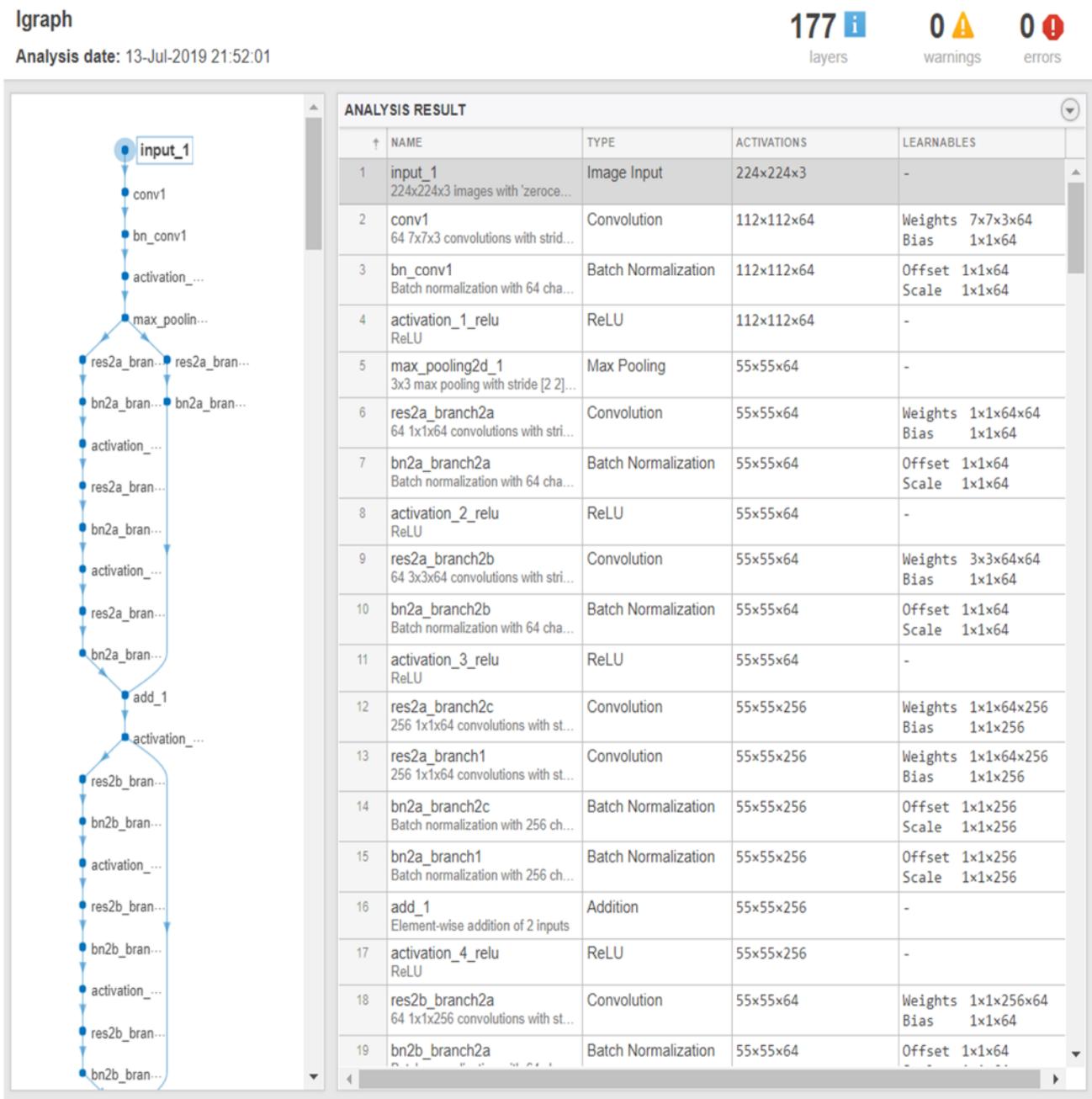


Figure 24: Architecture de ResNet 50 sous Matlab

6. Présentation de l'application :

On a conçu une application qui rend service à l'agriculteur et ceci en élaborant un programme qui permet de classer des images agricoles prises par un drone ou une caméra en deux classes selon leur taux de mauvaises herbes: classe B pour des images contenant un taux élevé de mauvaises herbes et classes A pour celles contenant moins de mauvaises d'herbes et selon la typologie de la plante (en tt on a 12 types de plantes) en utilisant le deep learning en utilisant deux modèles différents (deux architectures différentes) et à base de ces informations l'agriculteur prend les mesures nécessaires telle que la vaporisation des pesticides et des insecticides aux endroits concernés.

Ensuite afin de Mesurer les performances de chaque méthode de classification utilisé on a eu recours à la matrice de confusion et la courbe ROC.

Et les étapes suivies sont les suivantes :

- Préparation de la DATASET
- Utilisation des réseaux pré entraînés
- L'apprentissage du réseau
- Extraction des caractéristiques
- Création d'un classifieur
- Classification de nouvelles images

7. Les bases d'images :

On a principalement deux datasets la premier (dataset1) qui contient des images agricoles repartit en deux classes ,Classe A(classe des images ou il n y a pas beaucoup d'herbes) , Classe B (classe des images ou il y a beaucoup d'herbes) , le deuxième(dataset2) est formé de 12 classes de différents types de plantes agricoles classes (Ail ,Aubergine, Betterave ,Haricot Rouge ,Haricot vert ,Herbe noire ,Lentille ,Mais ,Moutarde des champs ,Poivrier noir , Salade , Tomate)et les images des deux datasets sont de format PNG .

7.1. Dataset1:

Dataset1 est une base de données d'images prises par un drone dans un champs agricole.

Ce dataset est partitionné en deux principales classes (A) qui est constitué d'images de plantes qui n'ont pas beaucoup de mauvaises herbes au contraire de la classe (B) qui elle en dispose d'images de plantes qui ont beaucoup de mauvaises herbes .

Le nombre d'images de chaque classe est de 100 et sont de format PNG de taille 1,76 Mo et de dimension 1296 *966 pixels .

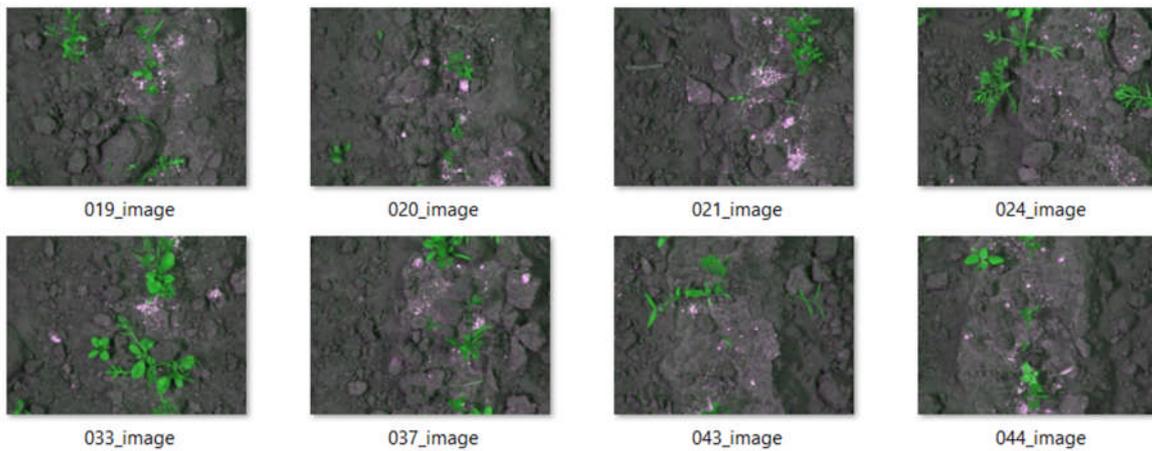


Figure 25: Echantillon d'images classe A.

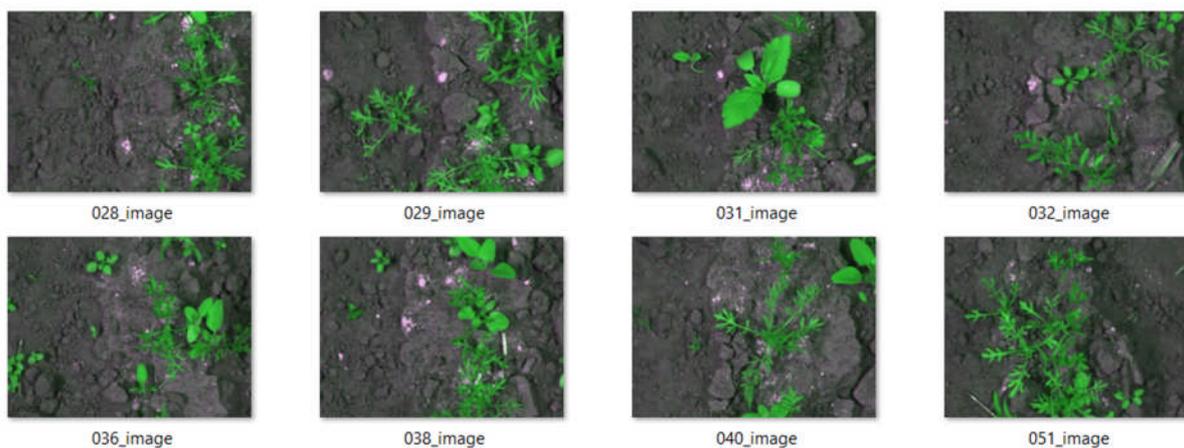


Figure 26: Echantillon d'image classe B.

7.2. Dataset 2 :

Dataset 2 est une base d'images qu'on a téléchargée a partir du net « V2-plant-seedling-dataset ». Elle contient 12 classes agricoles (Ail ,Aubergine, Betterave ,Haricot Rouge ,Haricot vert ,Herbe noire ,Lentille ,Mais ,Moutarde des champs ,Poivrier noir , Salade , Tomate) et chaque classe est disposée [100] images d'un type de plante bien définit de taille différentes mais majoritairement de taille 2.7 Mo et de dimension 1435*1435 pixels et pour que toutes les images soient de la même taille et avec les mêmes dimensions on a procédé à leur redimensionnement.

Voici quelques échantillons de deux classes Mais et Betterave parmi les douze classes :



Figure 27: Echantillon d'images de classe maïs.

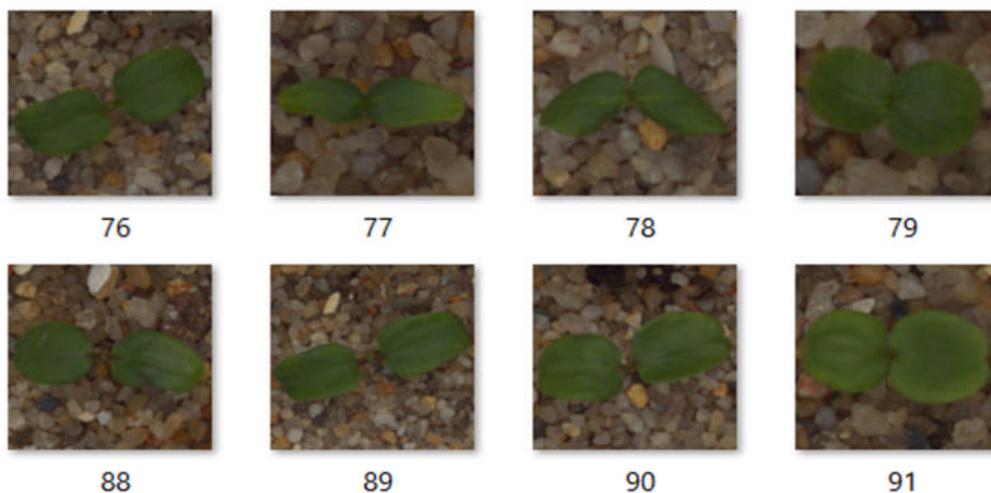


Figure 28 : Echantillon d'images classe betterave.

8. Pré traitement des données

Avant l'entraînement du réseau on doit passé par une étape importante qui est le découpage des deux bases de données (qui peut être aléatoire ou pas) en 2 voire 3 parties

Dans notre cas on a modélisé trois types différents de datasets :

- **Dataset d'entraînement.**
- **Dataset de validation.**
- **Dataset d'évaluation.**

La première va permettre à notre réseau d'apprendre et d'extraire des caractéristiques distinctes de chacune classe.

La seconde quand à elle va servir à valider le modèle en fin de chaque itération au cours de l'entraînement. En effet, en montrant de nouvelles images à notre réseau, il va lui permettre de se recalibrer pour éviter de sur-apprendre du jeu de données d'entraînement. Cette calibration va lui permettre d'avoir une meilleure généralisation de données.

Les données d'évaluation servent non pas à mesurer l'efficacité d'un algorithme comme celle de validation mais à comparer des algorithmes entre eux. Comme on l'a dit, il faut bien souvent tester plusieurs réseaux en faisant varier le nombre de couches, de neurones, d'étapes d'apprentissage, etc... Les données d'évaluation vont ainsi permettre de retenir le meilleur modèle au final (mais il faut les enlever AVANT d'apprendre).

Il faudra respecter un certain ratio entre ces deux jeux de données. A partir de notre dataset original, nous allons récupérer 70 à 80% des données pour le dataset d'entraînement, et donc de 10 à 20% pour le dataset de validation et d'évaluation

9. Entraînement du modèle

La partie la plus simple, ou il faut juste donner l'accès aux données vers réseau et attendre que notre réseau apprenne. Celui-ci va se renforcer au fur et a mesure des itérations que va parcourir notre modèle sur notre jeu de donnée, devenant ainsi meilleur.

Une fois le modèle entraîné, on va voir comment il s'est comporté durant l'entraînement,

Soit avec la fonction de perte ou de précision, on va pouvoir avoir de réels informations et indices sur le comportement de notre réseau, et ce sur le jeu de donnée d'entraînement et de validation.

La phase d'apprentissage de alexNet présente des caractéristiques selon ce qui suit :

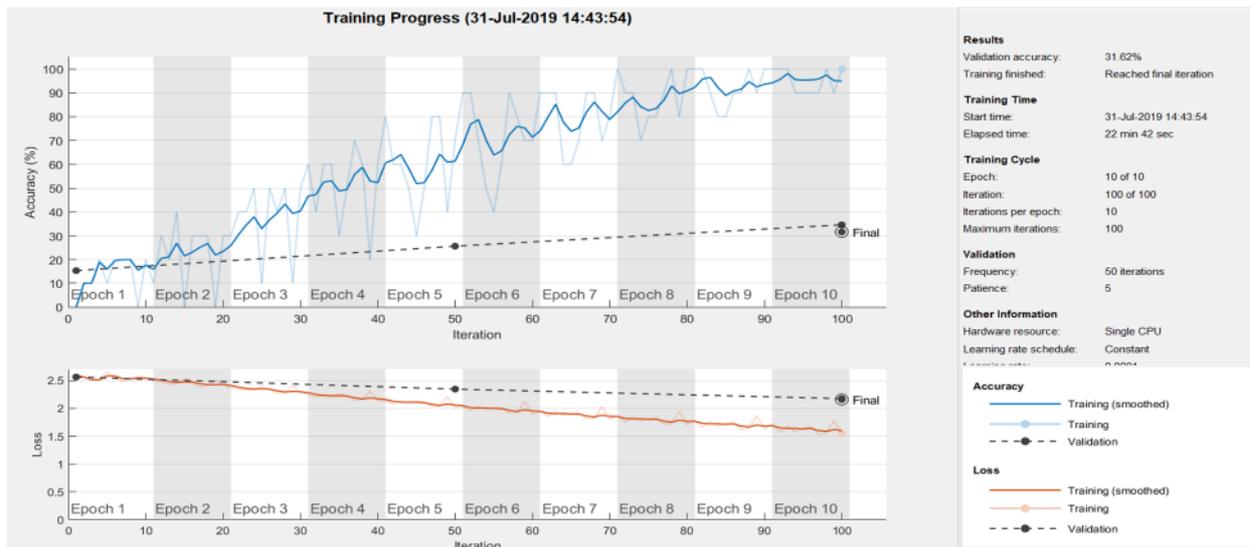


Figure 29 : graphe de suivit de l'entrainement de AlexNet.

A partir de la figure on remarque que le modèle n'a pas fini d'apprendre, de plus que la courbe montrant le jeu de données de validation reste stationnaire à partir des 30%.

Nous verrons un peu plus loin comment remédier à ça et améliorer notre modèle.

9.1. Sur-apprentissage et sous-apprentissage

A la fin du processus d'apprentissage trois cas de figures peuvent se présenter :

- Le modèle est performant sur les données d'entraînement (images sur lesquelles il s'entraîne) et aussi sur les données de validation (images qu'il n'a jamais vues). Cela est le cas **de figure idéale**, ça signifie que le modèle a très bien fait son travail et qu'il reconnaît aussi bien les images qu'il connaît que celles qu'il n'a jamais vues.
- Le modèle reconnaît très bien les images d'entraînement et moins bien celles de validation. Le modèle aura une faible capacité prédictive, il n'arrive pas à généraliser. On parle alors de **sur-entraînement**. Dans ce cas-là on peut ajouter d'avantage d'images pour pallier ce problème. (le soucis rencontré dans notre cas)
- Le modèle ne reconnaît pas très bien les images d'entraînements et pas très bien non plus les images de validation. On parle alors de **sous-apprentissage**. Dans ce cas ajouter plus d'images ne servira à rien, c'est généralement le modèle choisit qui ne convient pas, il faudrait utiliser un modèle plus complexe et avec plus de couches.

9.2. Visualisation des filtres Convnet :

Une autre façon de découvrir ce que le réseau de convolution recherche dans les images consiste à visualiser les filtres de couche de convolution. En affichant les filtres de couche réseau, nous pouvons en savoir plus sur le modèle .

On a vu des exemples de filtres connus ci-dessus. Cependant, dans la pratique, les filtres d'un réseau de neurones convolutif ont un kernel (noyau en français), stride, padding définis au préalable mais leurs valeurs sont générées aléatoirement à l'initialisation.

Ensuite, lorsque le réseau apprend, les valeurs des filtres sont mises à jour pour améliorer les résultats du CNN : les valeurs des filtres font donc parties des variables (poids, biais...) que le réseau change en apprenant

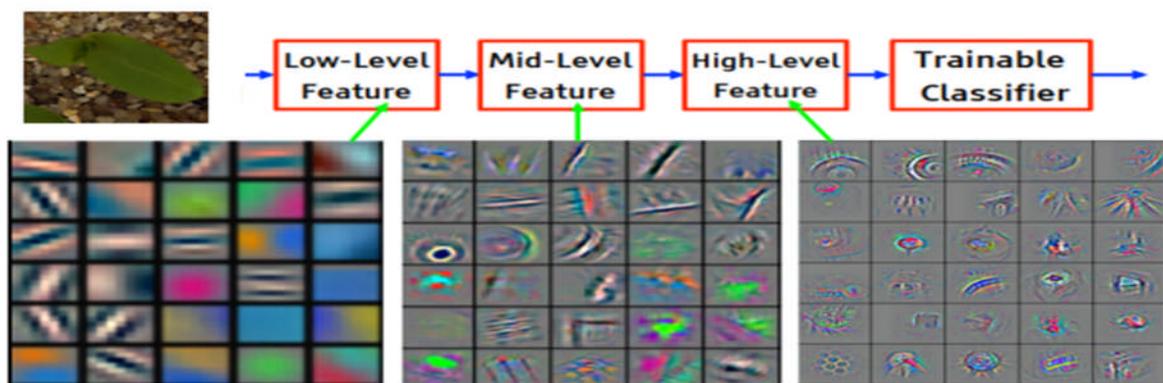


Figure 30: :exemples de filtres à plusieurs niveaux du CNN : au début, au milieu (après quelques convolutions) et à la fin

Après avoir examiné de plus près ces images de filtres de différentes couches de convolution, il devient très clair que les différentes couches cherchent réellement à apprendre des données d'image qui leur sont fournies. Les motifs trouvés dans les filtres dans les couches de départ semblent très basiques, composés de lignes et d'autres formes de base, ce qui indique que les couches précédentes apprennent les caractéristiques de base des images telles que les bords, les couleurs, etc. , les motifs deviennent plus complexes, ce qui suggère que les couches les plus profondes se renseignent sur des informations beaucoup plus abstraites, ce qui les aide à généraliser sur les classes et non sur l'image spécifique.

Et ce qui suit est une image du filtre de la première couche convolutive respectivement de AlexNet et ResNet50 :

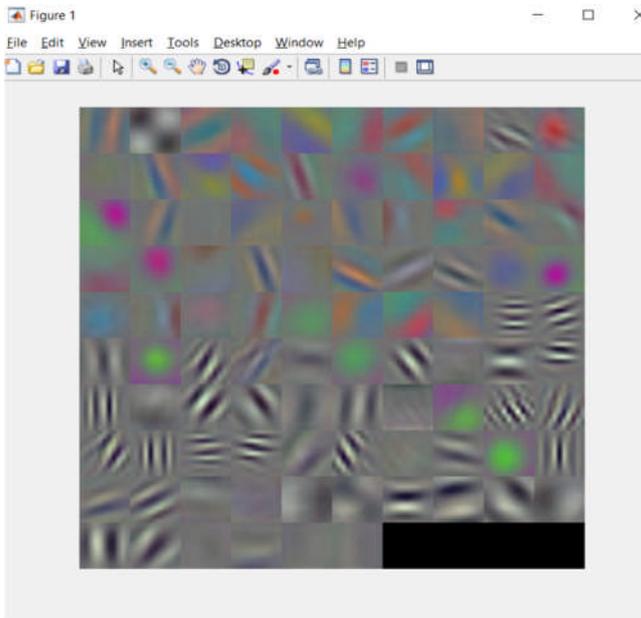


Figure 31: 96 filtres de la première couche de convolution de AlexNet.

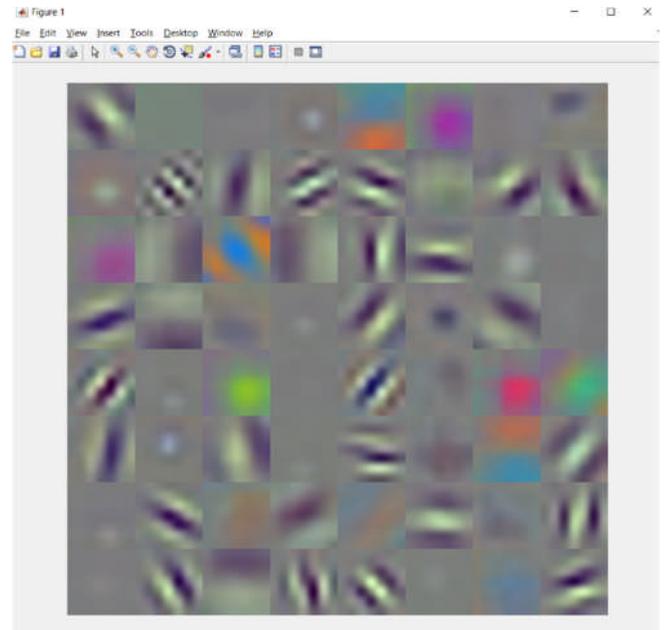


Figure 32: 64 filtres de la première couche de convolution de ResNet50.

Avant l'application du premier filtres les images sont redimensionnées à la taille attendue par le réseau, AlexNet (227*227), ResNet(224*244).

Ce qui suit est une application du filtre de la première couche de alexNet sur une image de maïs :

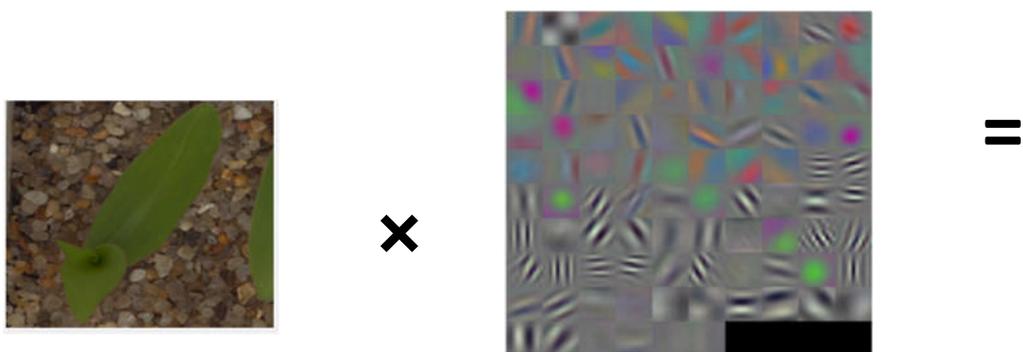


Figure 33 : Image d'entrée (227*227) et application d'un filtre AlexNet.

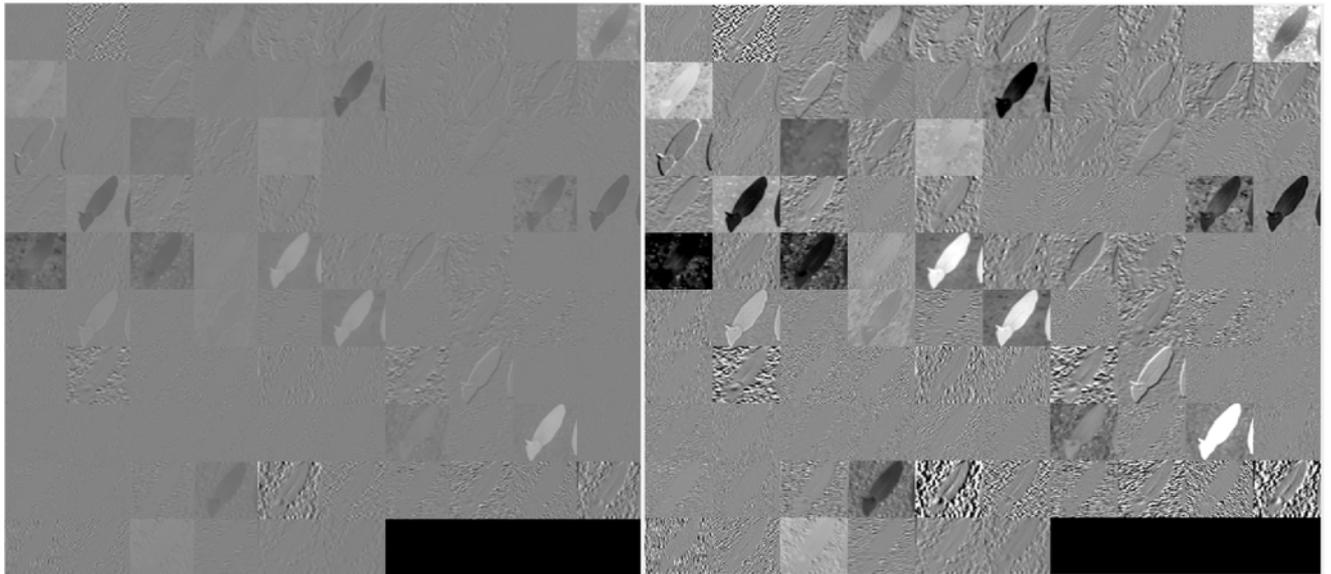


Figure 34 :résultat d'application du filtre et un étirement de contraste pour mieux visualiser.

Maintenant on s'intéresse à une autre couche, celle qui suit immédiatement, couche d'unité linéaire rectifiée(couche d'activation ReLU).

Une telle couche ne fait que changer tout nombre négatif à 0. Cela signifie que toute la variation des valeurs négatives de la sortie de la couche précédente est supprimée.

A quoi cela ressemble-t-il ?

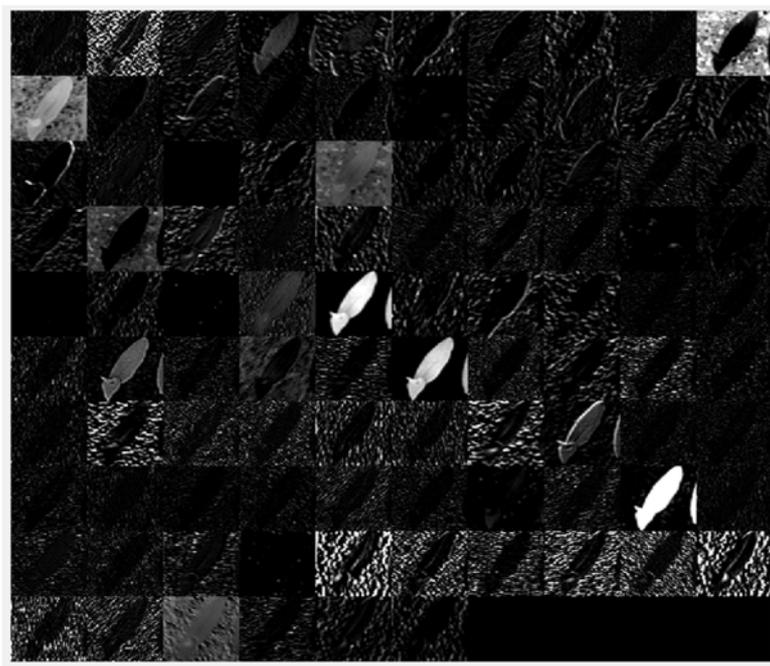


Figure 35:résultat d'application de la couche ReLu.

9.3.Axe d'amélioration :

On voit sur le graphique de suivi de notre courbe d'apprentissage laisse à désirer sur le jeu de données de validation, y a plusieurs pistes pour corriger cela et nous permettre de développer un modèle bien plus robuste.

Augmenter notre jeu de données : en effet, on a 100 images pour chacune de nos classes.

Plus on va fournir un jeu de données important et diversifié, plus il pourra apprendre et donc réaliser de meilleures prédictions.

Nous pouvons soit en récupérer davantage nous même à la main ,mais notre jeu de données est cependant limité ou impossible à étendre, nous avons choisis d' utiliser des techniques de **data augmentation**.

Augmenter la taille du réseau ,en augmentant notre jeu de données, nous allons pouvoir augmenter la profondeur de notre réseau de neurones. L' Ajout des couches va permettre au réseau d'extraire des caractéristiques plus complexes.

Augmenter la résolution de nos images d'entrées : n'ayant pas un GPU à disposition pour nos entraînements, on est dans l'obligation d'utiliser seulement un CPU, nous nous limitons ainsi dans les calculs. Cependant, augmenter la résolution des images va permettre au réseau de mieux s'en sortir. En effet, plus la qualité des images du dataset est haute, et plus les prédictions en seront bonnes.

9.4.C'est quoi Data augmentation ?

Pour pouvoir entrainer nos modèles, il nous faut d'énormes quantités de données. En effet, la quantité et surtout la qualité de notre dataset va avoir un rôle majeur pour l'élaboration d'un modèle de bonne facture. Il est logique d'avoir des données qui sont semblables entre elles c'est-à-dire qu'elles aient le même format, la même taille et longueur.. etc.

Et c'est à partir de ces contraintes que commencent les problèmes. En effet, avoir d'autres data spécifiques selon notre problème peuvent souvent relever de l'impossible. C'est là que la data augmentation va pouvoir nous être grandement utile.

Le principe de data augmentation repose sur le principe d'augmenter de façon artificielle nos données, en y appliquant des transformations. On va pouvoir augmenter la diversité et donc le champ d'apprentissage de notre modèle, qui va pouvoir mieux s'adapter pour comprendre de nouvelles données. Le principe de cette méthode est relativement simple, exemple :

- Addition de bruit
- Rotation
- Changement de couleurs
- Perte d'information
- Luminosité
- Contraste

Exemple en image :



Figure 3: Exemple d'augmentation avec rotation et ajout de contraste, luminosité à l'image original

En partant d'une simple image, nous pouvons la dupliquer autant de fois que nous avons des types de transformation différentes à lui appliquer. Et nous pouvons en augmenter davantage en croisant ces effets sur une même image, et en y appliquant différentes valeurs de l'effet.

Après amélioration

Après augmentation de la base de données du double et l'utilisation d'un réseau plus profond (avec plus de couches que le premier) (resnet 50) on a constaté que le taux d'apprentissage a considérablement augmenté, comme le montre les figures suivantes :

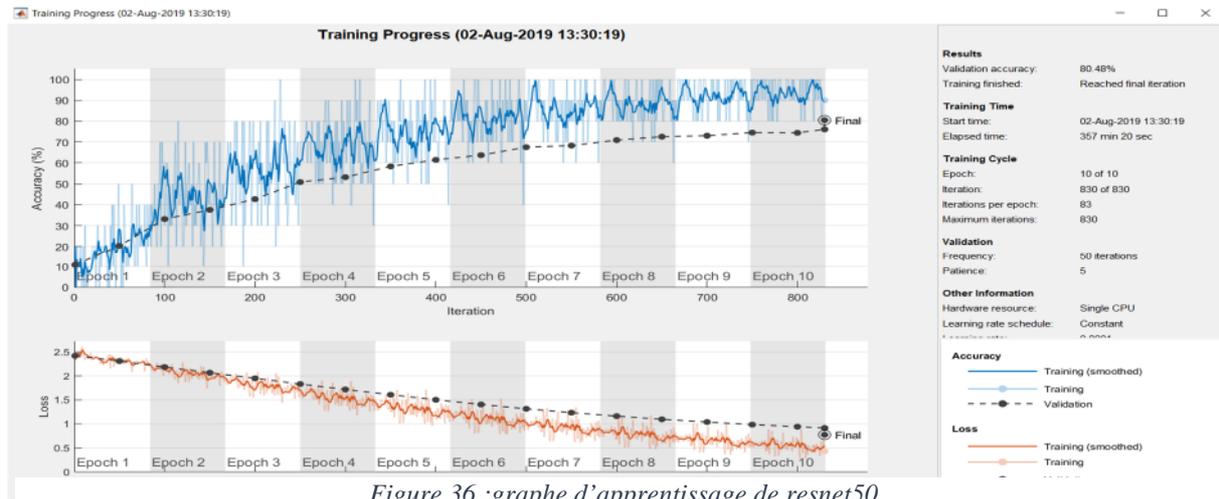


Figure 36 : graphe d'apprentissage de resnet50

Initializing image normalization.

Epoch	Iteration	Time Elapsed (hh:mm:ss)	Mini-batch Accuracy	Validation Accuracy	Mini-batch Loss	Validation Loss	Base Learning Rate
1	1	00:00:13	20.00%	10.73%	2.3267	2.4193	1.0000e-04
1	50	00:21:30	20.00%	20.29%	2.3342	2.3082	1.0000e-04
2	100	00:42:56	30.00%	33.13%	2.1746	2.1839	1.0000e-04
2	150	01:04:09	50.00%	37.54%	2.0269	2.0667	1.0000e-04
3	200	01:25:31	60.00%	42.68%	1.7607	1.9504	1.0000e-04
4	250	01:46:43	70.00%	50.85%	1.7362	1.8312	1.0000e-04
4	300	02:08:02	70.00%	53.21%	1.5390	1.7179	1.0000e-04
5	350	02:29:08	70.00%	58.35%	1.2572	1.6082	1.0000e-04
5	400	02:50:24	70.00%	61.48%	1.2692	1.5028	1.0000e-04
6	450	03:11:33	70.00%	63.74%	1.2833	1.4065	1.0000e-04
7	500	03:32:55	90.00%	67.54%	0.9595	1.3157	1.0000e-04
7	550	03:54:05	80.00%	68.31%	1.1185	1.2360	1.0000e-04
8	600	04:15:09	80.00%	70.93%	0.7395	1.1643	1.0000e-04
8	650	04:36:20	100.00%	72.57%	0.5887	1.0982	1.0000e-04
9	700	04:57:30	90.00%	73.04%	0.5930	1.0412	1.0000e-04
10	750	05:18:47	90.00%	74.52%	0.6332	0.9877	1.0000e-04
10	800	05:40:00	90.00%	74.42%	0.5727	0.9437	1.0000e-04
10	830	05:57:20	90.00%	76.07%	0.4340	0.9177	1.0000e-04

Figure 37 : Valeur de resnet50

D'après la Figure 36 : La précision de l'apprentissage et de validation augmente avec le nombre d'époques, ceci reflète qu'à chaque itération le modèle apprend plus d'informations. Si la précision a diminuée alors on aura besoin de plus d'informations à faire apprendre notre modèle. De même, l'erreur d'apprentissage et de la validation diminue avec le nombre d'époque

10. Matrice de confusion :

Une matrice confusion est un résumé des résultats de prédictions sur un problème de classification. Les prédictions correctes et incorrectes sont mises en lumière et réparties par classe. Les résultats sont ainsi comparés avec les valeurs réelles.

Pour calculer une matrice de confusion, il est nécessaire de disposer d'un ensemble de données de test (test dataset) ou d'un ensemble de données de validation avec les valeurs de résultat attendues. On fait ensuite une prédiction pour chaque ligne du testdataset

A partir des résultats escomptés et des prédictions, la matrice indique le nombre de prédictions correctes pour chaque classe et le nombre de prédictions incorrectes pour chaque classe organisée en fonction de la classe prédite. Chaque ligne du tableau correspond à une classe prédite, et chaque colonne correspond à une classe réelle

Dans les lignes sous les classes réelles, les prédictions ou les résultats sont inscrits. Ces résultats peuvent être l'indication correcte d'une prédiction positive comme « vraie positive » (true positive) et d'une prédiction négative comme « vraie négative » (true negative), ou une prédiction positive incorrecte comme « fausse positive » (false positive) et une prédiction négative incorrecte comme « fausse négative » (false negative).

L'avantage de ces matrices est qu'elles sont très simples à lire et à comprendre. Elles permettent de visualiser très rapidement les données et les statistiques afin d'analyser les performances d'un modèle et d'identifier les tendances qui peuvent aider à modifier les paramètres. Pour les problèmes de classification avec trois classes ou plus, il est aussi possible d'utiliser une Confusion Matrix en ajoutant des lignes et des colonnes.

10.1. Comprendre les terminologies de la Confusion Matrix (Matrice de confusion) :

Pour bien comprendre le fonctionnement d'une matrice de confusion, il convient de bien comprendre les quatre terminologies principales : TP, TN, FP et FN. Voici la définition précise de chacun de ces termes :

- **TP (True Positive)** : les cas où la prédiction est positive, et où la valeur réelle est effectivement positive. Exemple : le médecin vous annonce que vous êtes malade, et vous êtes bel et bien malade.
- **TN (True Negative)** : les cas où la prédiction est négative, et où la valeur réelle est effectivement négative. Exemple : le médecin vous annonce que vous n'êtes pas malade, et vous n'êtes effectivement pas malade.
- **FP (False Positive)** : les cas où la prédiction est positive, mais où la valeur réelle est négative. Exemple : le médecin vous annonce que vous êtes malade, mais vous n'êtes pas malade.
- **FN (False Negative)** : les cas où la prédiction est négative, mais où la valeur réelle est positive. Exemple : le médecin vous annonce que vous n'êtes pas malade, mais vous êtes malade.

Après toutes ces explications regardant les résultats de nous matrice après amélioration réalisé grâce à Matlab

Output Class	A	Ail	Aubergine	B	Bettrave	Haricot rouge	Haricot vert	Herbe noire	Lentille	Maïs	Moutarde des champs	Poivrier noir	Salade	Tomate cerise	
A	54 6.4%	0 0.0%	0 0.0%	4 0.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	93.1%
Ail	0 0.0%	48 5.7%	3 0.4%	0 0.0%	7 0.8%	15 1.8%	2 0.2%	27 3.2%	26 3.1%	3 0.4%	0 0.0%	2 0.2%	0 0.0%	6 0.7%	65.5%
Aubergine	0 0.0%	0 0.0%	19 2.3%	0 0.0%	0 0.1%	1 0.0%	0 0.0%	0 0.0%	1 0.1%	0 0.0%	0 0.0%	1 0.1%	2 0.2%	2 0.2%	73.1%
B	5 0.6%	0 0.0%	0 0.0%	56 6.7%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	91.8%
Bettrave	0 0.0%	0 0.0%	1 0.1%	0 0.0%	41 4.9%	0 0.1%	1 0.0%	0 0.0%	3 0.4%	7 0.8%	3 0.4%	0 0.0%	4 0.5%	0 0.0%	68.3%
Haricot rouge	0 0.0%	3 0.4%	0 0.0%	0 0.0%	1 0.1%	37 4.4%	0 0.0%	8 1.0%	2 0.2%	4 0.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	67.3%
Haricot vert	0 0.0%	0 0.0%	8 1.0%	0 0.0%	2 0.2%	1 0.1%	47 5.6%	0 0.0%	3 0.4%	1 0.1%	2 0.2%	2 0.2%	1 0.1%	1 0.1%	69.1%
Herbe noire	0 0.0%	6 0.7%	2 0.2%	0 0.0%	0 0.0%	4 0.5%	0 0.0%	25 3.0%	2 0.2%	0 0.0%	0 0.0%	1 0.1%	0 0.0%	0 0.0%	62.5%
Lentille	1 0.1%	0 0.0%	3 0.4%	0 0.0%	1 0.1%	0 0.0%	4 0.5%	0 0.0%	10 1.2%	0 0.0%	1 0.1%	0 0.0%	0 0.0%	0 0.0%	50.0%
Maïs	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 0.1%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	38 4.5%	0 0.0%	1 0.1%	0 0.0%	0 0.0%	95.0%
Moutarde des champs	0 0.0%	0 0.0%	4 0.5%	0 0.0%	1 0.1%	0 0.0%	0 0.0%	0 0.0%	4 0.5%	2 0.2%	48 5.7%	1 0.1%	3 0.4%	0 0.0%	76.2%
Poivrier noir	0 0.0%	0 0.0%	7 0.8%	0 0.0%	1 0.1%	1 0.1%	5 0.6%	0 0.0%	2 0.2%	1 0.1%	2 0.2%	48 5.7%	1 0.1%	2 0.2%	68.6%
Salade	0 0.0%	1 0.1%	3 0.4%	0 0.0%	4 0.5%	0 0.0%	0 0.0%	0 0.0%	4 0.5%	4 0.5%	4 0.5%	3 0.4%	44 5.2%	2 0.2%	63.8%
Tomate cerise	0 0.0%	2 0.2%	10 1.2%	0 0.0%	1 0.1%	1 0.1%	1 0.1%	0 0.0%	3 0.4%	0 0.0%	0 0.0%	1 0.1%	5 0.6%	47 5.6%	66.2%
	90.0%	80.0%	31.7%	93.3%	68.3%	61.7%	78.3%	41.7%	16.7%	63.3%	80.0%	80.0%	73.3%	78.3%	66.9%
	10.0%	20.0%	68.3%	6.7%	31.7%	38.3%	21.7%	58.3%	83.3%	36.7%	20.0%	20.0%	26.7%	21.7%	33.1%
	A	Ail	Aubergine	B	Bettrave	Haricot rouge	Haricot vert	Herbe noire	Lentille	Maïs	Moutarde des champs	Poivrier noir	Salade	Tomate cerise	
	Target Class														

Figure 38: matrice de confusion de AlexNet

Bien classé
 Mal classé
 Précision
 Total des échantillons

Comment on le voit sur cette figure les résultats montrent que le taux de classification erronée totale est de 33,1% (278 image mal classées) alors que le taux total classification juste est de 66,9% (562 images bien classées), ce qui tres bien mais on est loin pour dire que le réseau performant.

		Confusion Matrix														
Output Class	A	53 6.3%	0 0.0%	0 0.0%	1 0.1%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	98.1% 1.9%
	Ail	0 0.0%	50 6.0%	0 0.0%	0 0.0%	0 0.0%	1 0.1%	0 0.0%	24 2.9%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	3 0.4%	84.1% 35.9%
	Aubergine	0 0.0%	0 0.0%	32 3.8%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 0.1%	2 0.2%	1 0.1%	5 0.6%	78.0% 22.0%
	B	7 0.8%	0 0.0%	0 0.0%	59 7.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	89.4% 10.6%
	Bettrave	0 0.0%	0 0.0%	0 0.0%	0 0.0%	57 6.8%	2 0.2%	0 0.0%	1 0.1%	2 0.2%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	91.9% 8.1%
	Haricot rouge	0 0.0%	4 0.5%	1 0.1%	0 0.0%	2 0.2%	55 6.5%	1 0.1%	9 1.1%	5 0.6%	1 0.1%	0 0.0%	0 0.0%	0 0.0%	2 0.2%	68.8% 31.3%
	Haricot vert	0 0.0%	0 0.0%	4 0.5%	0 0.0%	1 0.1%	0 0.0%	50 6.0%	0 0.0%	3 0.4%	0 0.0%	0 0.0%	1 0.1%	0 0.0%	6 0.7%	76.9% 23.1%
	Herbe noire	0 0.0%	5 0.6%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 0.1%	20 2.4%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	76.9% 23.1%
	Lentille	0 0.0%	0 0.0%	2 0.2%	0 0.0%	0 0.0%	0 0.1%	2 0.2%	5 0.6%	45 5.4%	0 0.0%	3 0.4%	0 0.0%	0 0.0%	0 0.0%	77.6% 22.4%
	Mais	0 0.0%	1 0.1%	1 0.1%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 0.1%	2 0.2%	59 7.0%	0 0.0%	0 0.0%	0 0.0%	1 0.1%	90.8% 9.2%
	Moutarde des champs	0 0.0%	0 0.0%	1 0.1%	0 0.0%	0 0.0%	1 0.1%	0 0.0%	0 0.0%	3 0.4%	0 0.0%	56 6.7%	0 0.0%	0 0.0%	0 0.0%	91.8% 8.2%
	Poivrier noir	0 0.0%	0 0.0%	10 1.2%	0 0.0%	0 0.0%	0 0.0%	2 0.2%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	55 6.5%	0 0.0%	0 0.0%	82.1% 17.9%
	Salade	0 0.0%	0 0.0%	3 0.4%	0 0.0%	0 0.0%	0 0.0%	4 0.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	2 0.2%	59 7.0%	0 0.0%	86.8% 13.2%
	Tomate cerise	0 0.0%	0 0.0%	6 0.7%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	43 5.1%	87.8% 12.2%
		88.3% 11.7%	83.3% 16.7%	53.3% 46.7%	98.3% 1.7%	95.0% 5.0%	91.7% 8.3%	83.3% 16.7%	33.3% 66.7%	75.0% 25.0%	98.3% 1.7%	93.3% 6.7%	91.7% 8.3%	98.3% 1.7%	71.7% 28.3%	82.5% 17.5%
		A	Ail	Aubergine	B	Bettrave	Haricot rouge	Haricot vert	Herbe noire	Lentille	Mais	Moutarde des champs	Poivrier noir	Salade	Tomate cerise	
		Target Class														

Figure 39: matrice de confusion du modèle ResNet 50.

Bien classées
 Mal classées
 Précision
 Total des échantillons

D'après cette matrice le taux de précision de l'algorithme est de 82,5% donc 693 images bien classées sur les 840 images (des 30%), alors que le taux d'erreur est de 17,5%, ce qui fait 147 images mal classées sur les 30 %

D'une manière générale, un réseau de neurone CNN plus il est profond plus il donne de bons résultats et la de bonne performance comme on vient de voir avec ResNet50 et AlexNet

10.2. Réaliser une prédiction :

Maintenant que notre modèle est entraîné, on va enfin pouvoir réaliser des prédictions sur de nouvelles images. Et testant une nouvelle image qui est prise dans la base de données évaluation

Exemple en prend une image suivante (test1.png) qui est une image de maïs avec une dimension de 940*490



Ajoutant l'image dans le programme pour la tester est voir s'il va reconnaître.

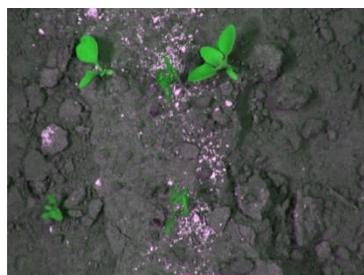
```
161 |
162 |
163 |
164 |
165 | %donner une image a testé
166 - newImage = imread(fullfile('test1.png'));
167 |
168 | % Créez augmentedImageDatastore pour redimensionner automatiquement l'image lorsque
169 | % les caractéristiques de l'image sont extraites à l'aide d'activations
170 - ds = augmentedImageDatastore (imageSize, newImage , 'ColorPreprocessing' , 'gray2rgb' );
171 | % Extraire les caractéristiques de l'image à l'aide de CNN
172 - imageFeatures= activations (net, ds, featureLayer, ...
173 | 'MiniBatchSize', 10, 'OutputAs', 'columns' );
174 | % Effectuer une prédiction à l'aide du classifieur
175 | Label = classify(net, newImage)
176 - Label = predict (classifier,imageFeatures , 'ObservationsIn', 'Columns' );
177 |
178 - sprintf('celle image appartient a la classe %s', Label)
179 |
180 |
181 |
182 |
183 |
184 |
```

Résultat du test :

```
ans =
'celle image appartient a la classe Maize'
```

Le programme a bien reconnu la classe de l'image

Essayant de prédire une autre image de type A (avec une densité de mauvaises herbes pas élevé) prise dans les images d'évaluation



Résultat d'évaluation ans =
 →
 'celle image appartient a la classe A'

Aussi dans cette deuxième prédiction le programme a bien trouvé la classe de l'image

Résumé des résultats :

Modèle	Précision obtenue sur la base d'apprentissage	Précision obtenue sur la base de validation	Temps d'exécution d'une prédiction	Précision des prédictions
AlexNet	100%	31%	2min	47%
ResNet	100%	65%	5 min 45s	59%

Tableau 2: Résultat avant augmentation du jeu de données.

Modèle	Précision obtenue sur la base d'apprentissage	Précision obtenue sur la base de validation	Temps d'exécution d'une prédiction	Précision des prédictions
AlexNet	90%	60%	5 min	68%
ResNet	100%	80%	16 min	78%

Tableau 3: Résultat après augmentation du jeu de données.

Les résultats obtenus sont exprimés en termes de précision d'apprentissage, de validation, de tests et erreurs et enfin de temps d'exécution. Le temps d'exécution est trop coûteux. Ceci revient à la grande dimension de la base ce qui nécessite l'utilisation d'un GPU au lieu d'un CPU.

Le tableau 2 a présenté les meilleurs résultats trouvés. Le nombre de couches de convolution reflètent ces bons résultats, cependant le temps d'exécution était très couteux.

Les résultats obtenus se sont améliorés à mesure que nous avons approfondie notre réseau et augmenté le jeu de données. La base d'apprentissage est également un élément déterminant dans les réseaux de neurones CNN, il faut avoir une base d'apprentissage de grande taille pour aboutir à des meilleurs résultats.

11.Courbe ROC :

Une courbe ROC (receiver operating characteristic) est un graphique représentant les performances d'un modèle de classification pour tous les seuils de classification. Cette courbe trace le taux de vrais positifs en fonction du taux de faux positifs :

Le taux de vrais positifs (TVP) est l'équivalent du rappel. Il est donc défini comme suit :

$$\text{TVP}=\text{VP}/\text{VP}+\text{FN}$$

Le taux de faux positifs (TFP) est défini comme suit :

$$\text{TFP}=\text{FP}/\text{FP}+\text{VN}$$

La représentation de roc met en avant un nouvel indicateur qui est l'aire sous la courbe, plus elle se rapproche de 1, plus le classifieur est performant .

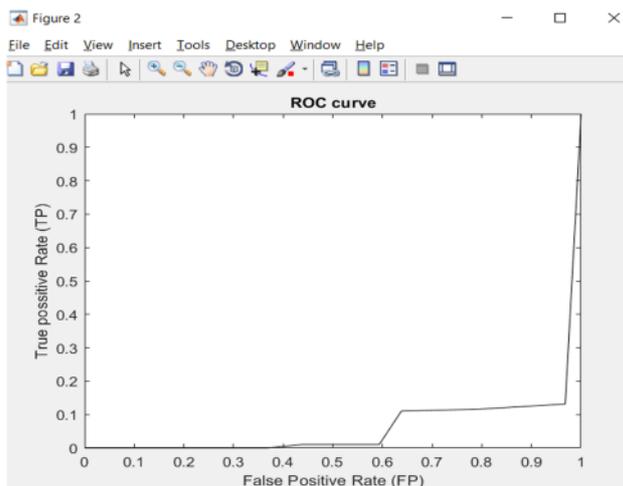


Figure 40: Courbe ROC de AlexNet avant amélioration.

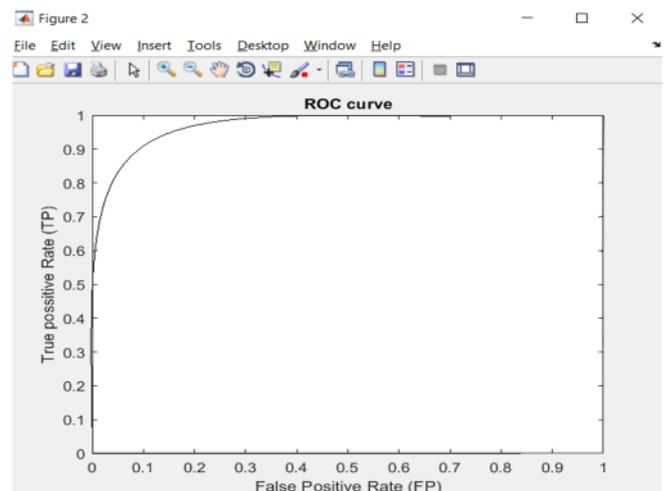


Figure 41: courbe ROC de ResNet 50 après amélioration.

12. Les principales lignes du code et leurs fonctionnalités :

Chargez le réseau neuronal pré-entraîné AlexNet. Si Deep Learning Toolbox TM *Modèle* pour AlexNet Network n'est pas installé, le logiciel fournit un lien de téléchargement

net= renvoie un réseau AlexNet pré-entraîné.

net = alexnet;

Afficher toutes les propriétés :

Layers =

25x1 [Layer](#) array with layers:

1	'data'	Image Input	227x227x3 images with 'zerocenter' normalization
2	'conv1'	Convolution	96 11x11x3 convolutions with stride [4 4] and padding [0 0 0 0]
3	'relu1'	ReLU	ReLU
4	'norm1'	Cross Channel Normalization	cross channel normalization with 5 channels per element
5	'pool1'	Max Pooling	3x3 max pooling with stride [2 2] and padding [0 0 0 0]
6	'conv2'	Convolution	256 5x5x48 convolutions with stride [1 1] and padding [2 2 2 2]
7	'relu2'	ReLU	ReLU
8	'norm2'	Cross Channel Normalization	cross channel normalization with 5 channels per element
9	'pool2'	Max Pooling	3x3 max pooling with stride [2 2] and padding [0 0 0 0]
10	'conv3'	Convolution	384 3x3x256 convolutions with stride [1 1] and padding [1 1 1 1]
11	'relu3'	ReLU	ReLU
12	'conv4'	Convolution	384 3x3x192 convolutions with stride [1 1] and padding [1 1 1 1]
13	'relu4'	ReLU	ReLU
14	'conv5'	Convolution	256 3x3x192 convolutions with stride [1 1] and padding [1 1 1 1]
15	'relu5'	ReLU	ReLU
16	'pool5'	Max Pooling	3x3 max pooling with stride [2 2] and padding [0 0 0 0]
17	'fc6'	Fully Connected	4096 fully connected layer
18	'relu6'	ReLU	ReLU
19	'drop6'	Dropout	50% dropout
20	'fc7'	Fully Connected	4096 fully connected layer
21	'relu7'	ReLU	ReLU
22	'drop7'	Dropout	50% dropout
23	'fc8'	Fully Connected	1000 fully connected layer
24	'prob'	Softmax	softmax
25	'output'	Classification Output	crossentropyex with 'tench' and 999 other classes

Regardons seulement les premiers couches, les hyper paramètres nous disent que cette couche effectue 96 opérations de filtrage différentes sur les canaux d'entrée, et chaque filtre est 11x11x3. La valeur de la foulée [4 4].

Les trois dernières couches du réseau pré-entraîné net sont configurées pour 1000 classes. Ces trois couches doivent être ajustées pour le nouveau problème de classification.

- **Modification du nombre de classe :**

Programme pour change le nombre de classe dans les 3 dernieres couches du réseau car la configuration initiale est à 1000 classes et dans notre cas deux classes :

```
layersTransfer = net.Layers (1: end-3);
numClasses = numel(categories)
layers = layersTransfer
fullyConnectedLayer(numClasses)
softmaxLayer
classificationLayer];
```

- **Résultat :**

```
numClasses =

    14
```

- **Découpage du jeu de données :**

On a divisé les ensembles en données d'apprentissage et de validation. 70% des images de chaque ensemble pour les données d'apprentissage et le reste, 30%, pour les données de validation.

```
[trainingSet, testSet] = splitEachLabel (imds, 0.7, 'randomize' );
```

- **Résultat :**

Nombre d'images pour l'apprentissage :

```
augmentedTrainingSet =
|
  augmentedImageDatastore with properties:

      NumObservations: 1960
```

Nombre d'image de validation :

```
augmentedTestSet =

  augmentedImageDatastore with properties:

      NumObservations: 840
```

- **Redimensionner les images :**

La première couche, la couche d'entrée d'image, nécessite des images d'entrée de taille 227 sur 227 sur 3, où 3 représente le nombre de canaux de couleur (RGB).

```
imageSize = net.Layers (1) .InputSize  
  
augmentedTrainingSet = augmentedImageDatastore(imageSize(1:2), trainingSet,  
'ColorPreprocessing' , 'gray2rgb' )  
  
augmentedTestSet = augmentedImageDatastore (imageSize(1:2), testSet,  
'ColorPreprocessing' , 'gray2rgb' )
```

- **Extraction des fonctionnalités d'apprentissage à l'aide de la couche 'fc8' :**

La fonction d'activation utilise automatiquement un GPU pour le traitement s'il en existe un, sinon un processeur est utilisé.

Dans le code ci-dessus, le paramètre 'MiniBatchSize' est défini sur 32 pour garantir que le CNN et les données d'image s'intègrent dans la mémoire du processeur.

```
featureLayer = 'fc8' ;  
trainingFeatures = activations (net, augmentedTrainingSet, featureLayer,  
...  
'MiniBatchSize' , 32, 'OutputAs' , 'columns' );
```

- **Formation d'un classifieur SVM multiclasse à l'aide des fonction CNN dans matlab :**

```
trainingLabels = trainingSet.Labels;  
classifieur = fitcecoc (trainingFeatures, trainingLabels, ...  
'Learners' , 'Linear' , 'Coding' , 'onevsall' , 'ObservationsIn' , 'Columns'  
);
```

- **Appliquer le classifieur formé sur une image de test (ou une image prise par un drone) :**

```
newImage = imread(fullfile('test7.png'));
```

- **Extraire les caractéristiques de l'image à l'aide du CNN :**

```
ds = augmentedImageDatastore (imageSize, testImage, 'ColorPreprocessing' ,  
'gray2rgb' );  
imageFeatures = activations (net, ds, featureLayer, 'OutputAs' , 'columns'  
);
```

- **Faire une prédiction en utilisant le classificateur :**

```
Label = Predict (classifier, imageFeatures, 'ObservationsIn' , 'Columns' )
```

- **Affichage du résultat de la prédiction :**

```
sprintf('celle image appartient à la classe %s', Label)
```

Conclusion

Nous avons présenté dans ce chapitre l'implémentation de l'approche de classification d'image agricole basée sur CNN, pour cela nous avons utilisé deux modèles d'architectures et plusieurs expériences et présenté différents résultats obtenus en termes de précision et d'erreur. La comparaison des résultats trouvés a la taille de la base et la profondeur de réseaux, sont des facteurs importants pour l'obtention de meilleurs résultats.

Conclusion générale :

Au cours de notre travail nous avons introduit les notions et les aspects fondamentaux de l'apprentissage profond (Deep Learning) et on a évoqué ses différentes utilisations ,ensuite on a abordé les CNN réseaux de neurones convolutif s'imposant comme l'architecture idéale vu sa simplicité et son efficacité pour répondre à la problématique que nous avons introduit au tout début de ce mémoire qui est la réalisation d'une application qui permet de traiter des images agricoles pour qu'elles fournissent l'information sur les cultures et le type de la plante .

La création d'un nouveau réseau de neurones convolutif est coûteuse (en terme matériel et de quantité de données annotées nécessaires et la difficulté de définir ses paramètres) c'est pour ces raisons qu'on a eu recours à des réseaux pré-entraînés qui sont disponible publiquement et qui nous permettent d'exploiter la puissance des CNN sans être un expert ils nous permettent la classification d'images agricoles et en tirer de chaque image des informations puissantes et fiables .

On a utilisé deux principaux modèles qui sont AlexNet et Resnet 50 et qui en ont respectivement 8 et 50 couches , ces deux réseaux pré-entraînés sont mise-en-œuvre en utilisant deux datasets .

Afin de mesurer l'efficacité de la classification faite par AlexNet et Resnet 50 on a utilisé deux indicateurs de performance la matrice de confusion et la courbe ROC .

Perspectives :

En Agriculture, le Deep Learning est déjà largement exploré et promet de nouvelles réponses aux questions posées par le travail avec et pour le vivant :

- des solutions pour limiter l'impact environnemental, pallier le manque de main d'oeuvre ou réduire la pénibilité du travail, en association avec les objets connectés, la robotique et le drone.

- des solutions d'assistance au quotidien et d'aide à la décision pour simplifier la vie des agriculteurs et de leurs conseillers...

Mais d'une autre part le Deep Learning peut être aussi utilisé pour résoudre d'autres problèmes que la reconnaissance de plantes ou la classification :

- Pouvoir reconnaître des plantes malades de plantes saines.
- Pouvoir détecter et/ou identifier des maladies sur une zone pour déterminer une stratégie de gestion (zonage).
- Pouvoir détecter et/ou identifier des maladies à certains stades pour agir au bon moment mais aussi détecter les mauvaises herbes.
- Optimisation du traitement et de la classification d'images en intégrant les paramètres de la qualité de service (QoS) comme la vitesse de traitement, la fiabilité du résultat
- Elaboration de nouveaux réseaux pré-entraînés plus performants et réduction du taux d'erreurs à des valeurs très minimales.
- Optimiser le potentiel de rendement. en modulant les interventions (semis, engrais, produits phytosanitaires) en fonction des zones d'hétérogénéité diagnostiquées à l'aide d'outils d'observation performants.
- La possibilité de détecter quelle dose de produit utiliser en fonction de la plante, du lieu, de la date, de la température etc et tout ça à distance

Références Bibliographique :

[1]<https://www.oracle.com/fr/cloud/deep-learning-intelligence-artificielle.html>

[2]:LIAKOS, Konstantinos G., BUSATO,Patrizia, PEARSON, Simon, BOCHTIS,Dionysis et MOSHOU, Dimitrios. Machine learning in Agriculture : areview. Sensors. août 2018.

[3]Mokri Mohammed Zakaria (03Juillet 2017).Classification des Images avec les Réseaux de Neurones Convolutionnels.Mémoire Master.Université Abou Bakr BelkaidTlemcen.

[4] Céline Deluzarche- Journaliste .intelligence artificielle et deeplearning,
<https://www.futura-sciences.com/tech/definitions/intelligence-artificielle-deep-learning-17262/> .

[5]Bastien L(16 Juillet 2018).DeepLearning ou Apprentissage profond :définition, concepts
<https://www.lebigdata.fr/deep-learning-definition> .

[6]ChesnerDesir.Classification Automatique d'Images.Application à l'Imagerie du Poumon Profond. Université de Rouen 2013.

[7]<https://agriculturedeprecision.wordpress.com>

[8] :SUPINFO International University ,Deeplearning, les fonction d'activation , par Florent SIMON , publié le 04/10/2018

[9] : site (Pensée Artificielle) ,Focus :le reseau de neurones convolutifs, par Lambert R. 11 janvier 2019

[10] Frédéric BARNAY (Juin 2001) . Mémoire de maitrise en association avec la chambre d'agriculture de Vaucluse. Université de Provence .

[11] .Philippe Zwaenepoel& Jean-Michel Le Bars (Décembre 1997). L'agriculture de précision .

[13]. Guillaume JOLY(14 janvier 2019). Les Horizons
<https://leshorizons.net/2019/01/14/cest-quoi-agriculture-de-precision>

[14]<https://www.studiosport.fr/guides/drones/tout-savoir-sur-les-drones.html> .

[15]Julien DEJASMin (9 Avril 2018).Intelligence Artificielle .
<https://www.natural-solutions.eu/blog/histoire-du-deep-learning> .

[16] MoualekDjaloul Youcef .Deep Learning pour la classification des images .Mémoire Master.Université de Tlemcen. (Juin 2016)

[17] <https://fr.wikipedia.org>

[18] BoughabaMohammed ,Boukhris Brahim (Juin2017).L'apprentissage profond pour la classification et la recherche d'images par le contenu. Mémoire Master.Université de Ouargla.

[20] <https://openclassrooms.com/fr/courses/4470531-classez-et-segmentez-des-donnees-visuelles/5083336-decouvrez-les-differentes-couches-dun-cnn>

[21] <https://bu.univ-ouargla.dz/master/pdf/BENNAMIA-GHARBI.pdf?idmemoire=28>

[22] Adrien Guille. Évaluation en classification supervisée. Université Lyon 2
http://mediamining.univ-lyon2.fr/people/guille/m4101/evaluation_supervisee.html

[23] <https://fr.mathworks.com> site officiel de Matlab

[24] SiddharthDas (16 Novembre 2017). CNN Architectures: LeNet, AlexNet, VGG, GoogLeNet, ResNet and more.