

UNIVERSITÉ DU QUÉBEC

MÉMOIRE PRÉSENTÉ À  
L'UNIVERSITÉ DU QUÉBEC À TROIS-RIVIÈRES

COMME EXIGENCE PARTIELLE  
DE LA MAÎTRISE EN MATHÉMATIQUES ET INFORMATIQUE  
APPLIQUÉES

PAR  
MOHAMED AMINE BOUHEDADJA

UTILISATION DE L'APPRENTISSAGE PROFOND POUR LA CLASSIFICATION DES  
MAUVAISES HERBES DANS LE BLEUET NAIN

AVRIL 2021

Université du Québec à Trois-Rivières

Service de la bibliothèque

Avertissement

L'auteur de ce mémoire ou de cette thèse a autorisé l'Université du Québec à Trois-Rivières à diffuser, à des fins non lucratives, une copie de son mémoire ou de sa thèse.

Cette diffusion n'entraîne pas une renonciation de la part de l'auteur à ses droits de propriété intellectuelle, incluant le droit d'auteur, sur ce mémoire ou cette thèse. Notamment, la reproduction ou la publication de la totalité ou d'une partie importante de ce mémoire ou de cette thèse requiert son autorisation.

## Remerciements

Je remercie toute personne qui m'a apporté un soutien moral durant mon cursus. Je remercie également mon directeur de recherche et les professeurs qui ont évalué mon travail.

## Résumé

La prolifération des mauvaises herbes dans les champs de bluet constitue un problème pour la récolte et la qualité du produit. Afin de remédier à ce problème, l'agriculture de précision est utilisée comme outil d'aide aux méthodes d'éradications, soit par pulvérisation d'herbicide ou par sarclage.

Pour permettre l'éradication efficiente des mauvaises herbes, plusieurs méthodes de détection des mauvaises herbes peuvent être utilisées, qui ont pour intérêt de discriminer les différentes espèces de mauvaises herbes sous certaines conditions. L'utilisation des indices de végétation par exemple qui consiste à utiliser des algorithmes capables de différencier des parties dans l'image grâce à une particularité de ses couleurs. D'autres méthodes utilisent la détection par seuillage avec la différence de formes pour subdiviser l'image d'une parcelle en régions distinctes (ex : sol, végétation).

Dans ce contexte, notre recherche a pour objectif de faciliter la classification des mauvaises herbes les plus répandues dans les champs de bluets.

Grâce aux expérimentations que nous avons menées, nous avons mis au point un classificateur basé sur un réseau de neurones convolutif, capable de déterminer avec une bonne précision l'espèce à laquelle appartient la mauvaise herbe parmi les quatre autres.

Mots clés : Convolutif ; Réseaux, Neurone ; Apprentissage ; Profond ; Détection ; Mauvaises ; Herbes.

## Abstract

Weeds proliferation in blueberry fields represents an issue for a good yield and product quality. In order to fix this problem, precision agriculture assists the eradication methods, whether they are by using chemical herbicides or weeding.

There are several methods of detecting weeds, they have the advantage of discriminating certain species of weeds under certain conditions. The use of vegetation indices, for example, which consists of using algorithms capable of differentiating parts in the image by looking for particularities of its colour ranges. Other methods use detection by thresholding with difference in shape to subdivide the image of a plot into distinct regions (e.g.: soil, vegetation).

In this context, the purpose of our study is to facilitate the classification of the most common weeds sorts in blueberry fields. Our experiments conducted to make a classifier based on convolutional neural network model that can determine with a good accuracy the type of weed among the four most answered in the fields of cornflowers.

Key words: Convolutional; Neural; Network; Deep; Learning; Weed; Detection.

## Table des matières

<b>Chapitre 1.</b>	Introduction .....	9
<b>Chapitre 2.</b>	État de l'art. ....	13
2.1.	Introduction.....	14
2.2.	L'acquisition d'image.....	14
2.2.1.	Les satellites .....	14
2.2.2.	Installation dédiée.....	15
2.2.3.	Les drones.....	15
2.2.4.	L'utilisation des téléphones intelligents.....	16
2.3.	Le prétraitement des images.....	16
2.3.1.	Images en niveaux de gris .....	17
2.4.	Les indices de végétation .....	18
2.4.1.	L'indice d'excès de vert ExG .....	18
2.4.2.	L'indice d'excès de rouge ExR.....	19
2.4.3.	L'indice d'extraction de végétation CIVE.....	20
2.5.	Les méthodes de seuillages.....	20
2.5.1.	La méthode d'Otsu .....	20
2.6.	La détection par apprentissage .....	21
2.6.1.	L'EASA (Environmentally Adaptive Segmentation Algorithm) .....	22
2.6.2.	SVM (Support Vector Machine).....	23
2.6.3.	Réseaux Bayésiens.....	25
2.6.4.	Les réseaux de neurones artificiels .....	26
2.7.	Conclusion .....	28
<b>Chapitre 3.</b>	Cadre de recherche .....	29
3.1.	Introduction.....	30
3.2.	Les mauvaises herbes.....	30
3.2.1.	La comptonie voyageuse.....	30
3.2.2.	La kalmia à feuilles étroites .....	31
3.2.3.	Le quatre-temps .....	31
3.2.4.	Les touffes d'herbes .....	32
3.3.	L'apprentissage profond avec les réseaux de neurones convolutif (CNN) .....	32
3.3.1.	Le choix des CNN pour notre cas d'étude .....	33
3.3.2.	L'origine des CNN .....	33
3.3.3.	Architecture des CNN.....	34
3.3.4.	Exemples d'architectures de CNN.....	37

3.4.	Les outils de développement utilisés .....	40
3.4.1.	Outils et technologies.....	40
3.4.1.2.	Google Colaboratory .....	41
3.4.2.	Le choix du langage de programmation .....	41
3.4.3.	Librairies et Framework.....	41
3.5.	Conclusion .....	43
<b>Chapitre 4.</b>	<b>Méthodologie.....</b>	<b>44</b>
4.1.	Plan de travail.....	45
4.2.	Acquisition des images .....	45
4.3.	Organisation des données.....	46
4.3.1.	Le choix des images.....	48
4.3.3.	Taille du lot de traitement.....	49
4.3.4.	Nombre d'itérations .....	49
<b>Chapitre 5.</b>	<b>Résultats et discussion .....</b>	<b>54</b>
5.1.	Les métriques de performances.....	55
5.1.1.	L'exactitude de prédiction.....	55
5.1.2.	La perte d'entropie croisée catégorique .....	56
5.1.3.	La précision et le rappel et F1 .....	57
5.1.4.	La matrice de confusion .....	58
5.2.	Discussion .....	59
5.3.	Conclusion .....	59
<b>Chapitre 6.</b>	<b>Conclusion et perspectives .....</b>	<b>61</b>
	Bibliographie.....	64

## Table des figures

Figure 1 Exemple d'un système d'acquisition d'images.....	15
Figure 2 Exemple de l'utilisation d'un téléphone cellulaire avec un dispositif et mesure NDVI. Source (Chung et al., 2018).....	16
Figure 3 Application de la transformation RGB (image gauche) à niveau de gris (image droite). .....	17
Figure 4 Application de la transformation RGB à niveau de gris en suivant les recommandations UIT.....	18
Figure 5 Application de l'indice ExG sur une image RGB.....	19
Figure 6 Application de l'indice ExR sur une image RGB.....	19
Figure 7 Le système de traitement d'image.....	21
Figure 8 Organigramme de l'algorithme EASA, comprenant l'apprentissage, le calcul du classificateur et la génération de table de consultation. Source (Ruiz-Ruiz, Gómez-Gil, & Navas-Gracia, 2009).....	23
Figure 9 Illustration de séparation de deux classes avec SVM Linéaire à marge maximale. Source :(Wikipedia, 2020) .....	24
Figure 10 Ajout de propriété dans un SVM. Source : (Géron, 2019) .....	24
Figure 11 Classification SVM magenta : mauvaises herbes ; cyan : végétation. Source : (Pulido, Solaque et al. 2017) .....	25
Figure 12 Réseaux de neurones probabilistes. Source : (Chen, Zhang, Su, & Guo, 2010).....	27
Figure 13 La Comptonie voyageuse au printemps.....	31
Figure 14 Kalmia feuille verte, Image prise en été. ....	31
Figure 15 Kalmia feuille rouge, image prise en automne. ....	31
Figure 16 Quatre-Temps vert. Source : (repertoirequebecnature) .....	32
Figure 17 Quatre-Temps rouge. Source : (repertoirequebecnature) .....	32
Figure 18 touffes d'herbes prises lors d'une mission.....	32
Figure 19 Interconnexions entre les couches dans le Neocognitron . Source : (Fukushima, 1980).....	34
Figure 20 Architecture standard d'un réseau à convolutions. Source : (Wikipédia) .....	34
Figure 21 Couches de convolutions avec les récepteurs locaux en rectangle. Source (Géron, 2019).....	35
Figure 22 Max pooling avec un filtre $2 \times 2$ et un pas de 2. Source : (Wikipédia) .....	36
Figure 23 Graphe de la fonction ReLu.....	36
Figure 24 Progression des résultats des architectures CNN. Source : image-net.org .....	39
Figure 25 Illustration du modèle YOLO. Source :(Joseph Redmon et al., 2016).....	40
Figure 26 Un exemple d'utilisation de Numpy. Source : (Numpy.org) .....	42
Figure 27 Illustration de schéma avec Matplotlib. Source : (Matplotlib.org).....	43
Figure 28 Les étapes de notre plan de travail.....	45
Figure 29 Comptonie prise 25/10/2019.....	46
Figure 30 Exemple d'image de Kalmia, quatre-temps, touffe (tussock) et Comptonie. ....	47
Figure 31 Organisation des données .....	47
Figure 32 Diagramme du modèle avec les différentes couches le constituant.....	51
Figure 33 Caractéristiques extraites de la couche d'entrée des différentes espèces de mauvaises herbes.....	52
Figure 34 Résultats des réajustements du modèle.....	55
Figure 35 L'exactitude (à gauche) et la perte (à droite) pour l'entraînement et la validation. .	56

Figure 36 entraînement d'un modèle avec 50 passages ..... 57

---

## **Chapitre 1.** Introduction

---

Le traitement de données connaît un grand essor il impacte plusieurs filières dans la recherche ou le monde professionnel. Le domaine agricole n'échappe pas à cette réalité notamment avec l'agriculture de précision. Ainsi l'utilisation des nouvelles technologies est au sein du processus de bout en bout, de façon directe ou indirecte. La disponibilité et la facilité d'avoir de plus en plus de données à traiter, notamment d'images rendu facile à cause de la démocratisation des moyens de captures telles que les appareils photo, les drones, et surtout les téléphones cellulaires, ont grandement contribué au développement de moyens permettant leurs exploitations.

Plusieurs avenues ont été explorées afin de traiter les images capturées pour pouvoir en extraire des données exploitables pour l'amélioration de la productivité quantitative et qualitative. Notre recherche se focalisera sur un des problèmes majeurs de l'agriculture qui est la prolifération des mauvaises herbes dans les champs agricoles, la culture de bleuets nains dans le contexte de notre recherche. Les dernières avancées en termes de traitement et d'analyse d'images sont utilisées afin d'optimiser la discrimination des mauvaises herbes et des cultures tout en classifiant ces mauvaises herbes. La détection préalable et mieux ciblée des mauvaises herbes pourra ainsi jouer un grand rôle pour la promotion d'une culture biologique et plus écologique, car cela permettra une nette diminution de l'utilisation des herbicides dans le cas d'une culture conventionnelle ou alors le sarclage manuel ou automatiser dans le cas d'une culture biologique.

L'acquisition d'images requiert des équipements et des technicités spécifiques selon le cas et le but de la recherche, en tenant compte des avancées contemporaines de ces technologies. Parmi les outils les plus exploités, on peut citer les satellites avec leurs potentiels de couverture de larges zones et leurs capacités de fournir des images de différents types : spatiale, spectrale, temporelle, radiométrique et géométrique (Campbell. & Wynne., 2011). Un autre dispositif de plus en plus utiliser ces dernières années ce sont les drones, grâce à leur démocratisation et la baisse de leurs coûts, ils sont devenus un outil important dans la recherche en agriculture de pointes grâce à leur capacité de capture d'images contenant de multitudes données comme les images multispectrales qui permettent d'avoir des images incluant les bandes spectrales non perceptibles à l'œil humain.

L'image numérique contient une multitude de données, les formats d'images diffèrent entre autres par le choix des bandes spectrales utilisées pour les prendre, afin d'exploiter au mieux leur potentiel. Le plus connu est l'espace RVB (RGB en Anglais) qui est majoritairement utilisé en informatique ainsi qu'en imagerie, alors qu'en imprimerie on utilise le CMYK (Cyan, Magenta, Yellow et Black). Une autre représentation plus proche de la perception humaine des

couleurs, est l'espace TSL (Teinte, Saturation, Luminance ou Valeur) HSV en Anglais. En plus de ces représentations, dont les besoins de données dépassent les limitations de la perception humaine, on parle alors d'images multispectrales. Il s'agit de différentes bandes spectrales, en plus des couleurs de bases RVB, il existe d'autres bandes telles que les NIR (proche infrarouge), MID (moyen infrarouge) et FAR (loin infrarouge), ces trois dernières sont particulièrement utilisées dans l'imagerie de la végétation.

Pour extraire ou exploiter la prépondérance d'une couleur dans une image, l'utilisation des indices couleur est souvent le moyen le plus simple et également le plus efficace. Parmi ces indices il y'a l'ExG (indice d'excès du vert) ou l'ExR (indice d'excès du rouge) ainsi que plusieurs autres que nous détaillerons dans le second chapitre.

Plusieurs méthodes de discriminations des mauvaises herbes peuvent être utilisées, ces méthodes existent depuis longtemps, mais leurs utilisations dans le domaine agricole restent néanmoins récentes. Différentes approches ont été développées comme l'utilisation des machines à vecteurs de support (SVM) (Pulido, Solaque, & Velasco, 2017), les réseaux bayésiens (De Rainville et al., 2012). Dans notre recherche, nous nous focaliserons sur l'apprentissage avec les réseaux de neurones convolutifs qui à l'heure actuelle connaît une forte poussée et un regain d'intérêt énorme. Nous expliquerons par la suite les principaux facteurs ayant impactés leurs récentes popularisations.

Le présent document a pour objectif de décrire en détail la technique de classification des mauvaises herbes dans les champs de bluets utilisant les réseaux de neurones convolutifs (CNN) et sera structuré de la façon suivante :

Dans le chapitre 2, sur l'état de l'art, nous allons présenter de façon séquentielle les étapes du processus de classification suivies dans les recherches similaires pour permettre la détection et la classification des mauvaises herbes. À commencer par l'acquisition des images puis le prétraitement de celles-ci. Ensuite, les différentes méthodes de classification automatique utilisées dans le domaine de la vision par ordinateurs telles que les SVM, les réseaux bayésiens, ainsi que les réseaux de neurones. Le chapitre 2 se termine par une introduction sur les réseaux de neurones convolutifs (CNN) qui seront plus détaillés dans le chapitre suivant.

Dans le troisième chapitre, nous discuterons de la méthodologie appliquée dans notre recherche. Le but étant d'explorer les méthodes expérimentales utilisées dans notre recherche. Nous allons décrire en détail les différentes phases de classification des mauvaises herbes dans les champs de bluets. La capture des images multispectrales de mauvaises herbes et du bluet sera d'abord abordée. Ensuite, le processus de classification par réseaux de neurones

convolutifs (CNN) sera décrit. La conception des couches de ces CNN, leurs entraînements et leurs validations seront alors présentés.

Au chapitre 4, les résultats des différentes simulations seront présentés. Nous avons utilisé différents paramètres expérimentaux dont les résultats de chaque étape seront décrits avec une analyse détaillée.

Puis au chapitre 5, nous présenterons une conclusion ou nous allons parler des potentielles perspectives pour faire évoluer cette recherche et l'enrichir.

---

## Chapitre 2. État de l'art.

---

## 2.1. Introduction

Ce chapitre consiste en une exposition des différentes méthodes de traitement et d'analyse d'images appliquées en particulier dans l'agriculture de précision. Il s'agit ici de procédures permettant de traiter, discriminer ou de classifier les images en utilisant plusieurs algorithmes spécialisés.

Nous exposerons en premier, les techniques de segmentation d'images de plantes qui permettent de discriminer dans une image ce qui est de la végétation de ce qui ne l'est pas. Après nous aborderons différentes méthodes de classification d'images de plantes de différentes espèces. Puis, nous ferons référence aux recherches utilisées dans la détection et la discrimination des mauvaises herbes dans différentes cultures. Tout en abordant les termes utilisés dans le domaine du traitement d'image en particulier dans le domaine agricole.

Mais avant d'entamer ce chapitre plus en détail, nous présenterons succinctement les moyens qui permettent de capturer les images, en effet, il existe une multitude de techniques qui ont vu une énorme progression depuis plusieurs années gagnant en précision et surtout en se démocratisant, devenant plus abordable et permettant de prendre des images de plus en plus précises et plus facilement.

## 2.2. L'acquisition d'image

La capture d'images ou de vidéos dans le domaine agricole a progressé parallèlement à l'imagerie globalement. Si au début ce n'était qu'avec des appareils spécialement dédiés et qualitativement limités, les avancées qu'ont connues la prise de photos ou la capture d'images se sont améliorées et adaptées aux besoins dans les différentes sphères d'applications. Il est également indéniable que la démocratisation de cet outil est un des plus remarquables développements à ce niveau.

### 2.2.1. Les satellites

L'utilisation des satellites permet par exemple d'avoir une cartographie des champs ainsi qu'une géo référenciation afin d'obtenir des informations utiles dans divers domaines de recherche. Ils sont notamment utilisés dans des études nécessitant une large couverture de zones agricoles ou pour l'étude de la diversité de la flore, par exemple (Luoto, Toivonen, & Heikkinen, 2002).

### 2.2.2. Installation dédiée

Un autre concept a également fait part de son utilité dans le traitement de sujets plus particulier comme le traitement des grains tant pour la consommation animale ou humaine que pour l'ensemencement (graines de semence). Il est question de système de capture de photos pour la surveillance de l'évolution des graines pendant leurs phases de croissance ou alors pour la discrimination des bonnes et mauvaises graines basée par exemple sur leur potentielle de germination. Un exemple de ce genre d'outil est schématisé dans la Figure 1.

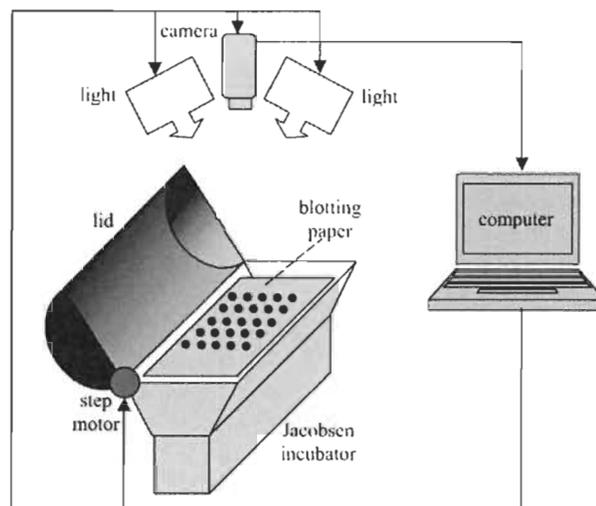


Figure 1 Exemple d'un système d'acquisition d'images.  
Source : (Ducournau et al., 2004)

### 2.2.3. Les drones

Plus récemment, la démocratisation et la régularisation de l'utilisation des véhicules aériens motorisés (UAV) (Ballesteros, Ortega, Hernández, & Moreno) ou plus communément appelés drone, à beaucoup bénéficié de la recherche dans l'agriculture de précision, en effet, ces véhicules permettent de capturer des images de grandes précisions. Les drones permettent notamment une énorme réduction des coûts comparés à l'utilisation des satellites et aussi une meilleure résolution des images de l'ordre du centimètre et même du millimètre (UAV) par rapport aux mètres pour les images satellitaires.

Selon le contexte de l'étude et l'exigence du contexte, plusieurs modes d'images peuvent être obtenues selon les caméras et les paramètres utilisés (image couleur RGB, images en niveau de gris, images en infrarouge et thermiques...etc.).

Une excellente application des drones permet de combiner leurs caméras et capteurs avec des techniques et algorithmes permettant de déduire des indices de végétation tels que le

ExG (Excess of Green) ou NDVI (Normalized Differential Vegetation Index) que nous allons aborder un peu plus loin dans ce chapitre. Cela dit il y a malgré tous des inconvénients à leurs utilisations, même si leur coût est bien moins élevé que celui des satellites, les drones de bonne qualité avec les bons équipements reviennent relativement chers. Ils ne sont pas pleinement autonomes et nécessite une bonne connaissance technique pour les piloter et pour parfois finir avec des informations acquises qui ne sont pas très exploitables (Ballesteros et al., 2014).

#### 2.2.4. L'utilisation des téléphones intelligents

Aujourd'hui l'objet technologique le plus vendu est sans nul doute le téléphone intelligent. Ces appareils embarquent des appareils photo avec des capteurs de plus en plus puissants. Vouloir tirer profit de cette puissance est donc naturel. Ainsi les téléphones cellulaires sont les parfaits substituts d'autres équipements pour certaines tâches de captures d'images et de contrôles utilisant leurs autres avantages tels que les GPS intégrés, leurs connectiques (GSM, WIFI, Bluetooth...etc.). De plus on peut également les augmenter avec d'autres dispositifs tels des filtres spécialisés, comme dans le cas d'une étude du stress impactant les plantes avec la déshydratation ou d'autres causes, où les chercheurs ont monté un filtre et calculé les valeurs NDVI des plantes pour extraire les informations pertinentes à l'étude. (Chung, Breshears, & Yoon, 2018). (Voir Figure 2)

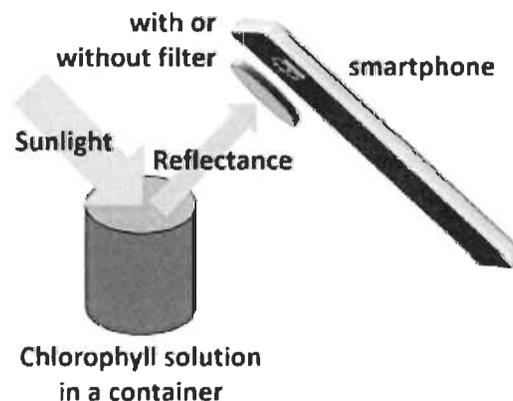


Figure 2 Exemple de l'utilisation d'un téléphone cellulaire avec un dispositif et mesure NDVI.  
Source (Chung et al., 2018)

### 2.3. Le prétraitement des images

Après la prise d'images, celles-ci doivent subir un prétraitement avant d'être exploitées. Ce prétraitement diffère selon le cas d'utilisation, les conditions de capture, les images obtenues et les résultats voulus.

### 2.3.1. Images en niveaux de gris

L'un des prétraitements les plus utilisés est la transformation d'image couleur (mode RGB) en images noir et blanc (niveau de gris) cela pouvant être utile à plusieurs fins, notamment la diminution de la taille de l'image, ce prétraitement est utilisé dans le cas où les couleurs ne font pas partie des caractéristiques utiles dans le traitement ultérieur. La méthode la plus simple pour l'appliquer est de prendre la moyenne des trois couleurs (Rouge, bleu et vert) de chaque pixel de l'image telle que :

$$Image\ en\ niveau\ de\ gris = \frac{R+G+B}{3}.$$

Mais cette formule ne donne pas le meilleur rendu d'une image comme on peut le voir dans la Figure 3.

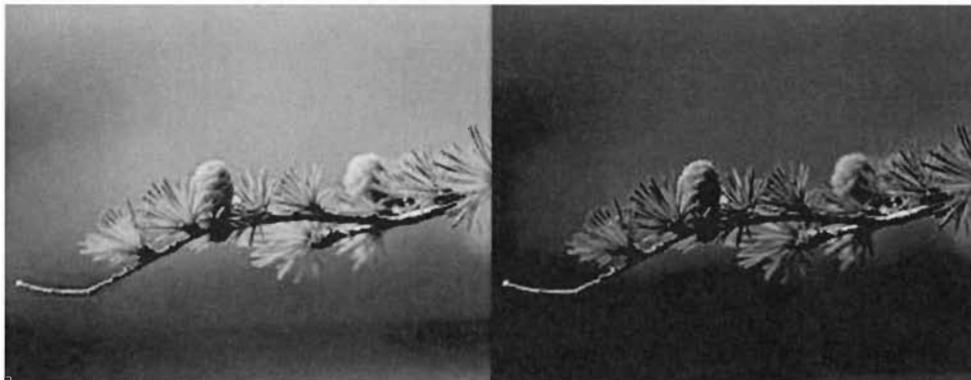


Figure 3 Application de la transformation RGB (image gauche) à niveau de gris (image droite).

Mais si on normalise les 3 couleurs du mode RGB en utilisant les recommandations de l'Union International des Télécommunications (UIT-R, 2011) données par la formule suivante :

$$image = ( 0.299 * Rouge + 0.587 * Bleu + 0.114 * Vert )$$

Le résultat obtenu est plus précis comme nous pouvons le voir dans la Figure 4.

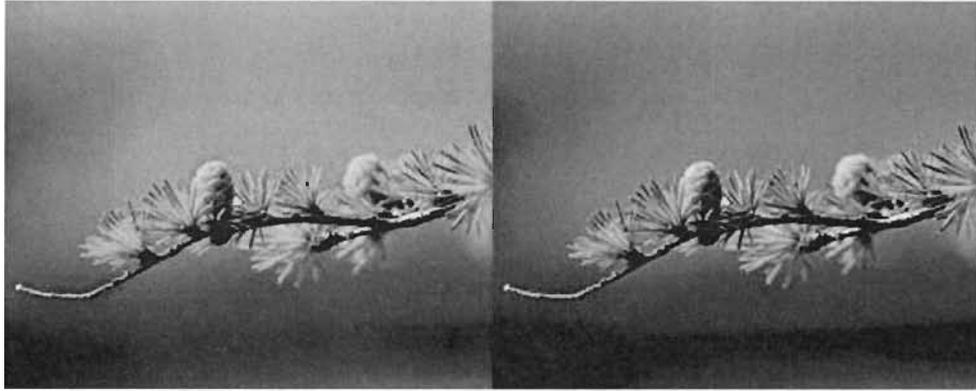


Figure 4 Application de la transformation RGB à niveau de gris en suivant les recommandations UIT

## 2.4. Les indices de végétation

Les indices végétation sont des moyens indispensables pour discriminer une plante du sol ou d'une autre plante. Il existe plusieurs indices de végétations couleurs, et nous n'allons pas faire une liste exhaustive ici, mais citer les plus populaires et utilisés.

### 2.4.1. L'indice d'excès de vert ExG

Afin de pouvoir extraire les caractéristiques des feuilles des plantes de façon optimale, il est possible d'exploiter une caractéristique propre à la végétation, soit de refléter la couleur verte grâce au pigment assimilateur chlorophylle. L'ExG est donc un excellent moyen en donnant comme résultat une image en niveau de gris où les zones vertes sont parfaitement discriminées (Meyer, Hindman, & Laksmi, 1999). La formule qui calcule cet indice est la suivante :

$$ExG = 2g - r - b$$

Où  $r$ ,  $g$  et  $b$  sont les coordonnées chromatiques :

$$r = \frac{R^*}{R^*+G^*+B^*} \quad g = \frac{G^*}{R^*+G^*+B^*} \quad b = \frac{B^*}{R^*+G^*+B^*}$$

$R^*$ ,  $G^*$  et  $B^*$  sont les valeurs normalisées des couleurs RGB de chaque pixel dans l'intervalle 0 à 1 :

$$R^* = \frac{R}{R_{max}} \quad G^* = \frac{G}{G_{max}} \quad B^* = \frac{B}{B_{max}}$$

$R$ ,  $G$  et  $B$  sont les valeurs de chaque composante de couleur des pixels dans une image respectivement le rouge, vert et bleu et où  $R_{max} = G_{max} = B_{max} = 255$  pour une image de 24 bits (voir Figure 5).

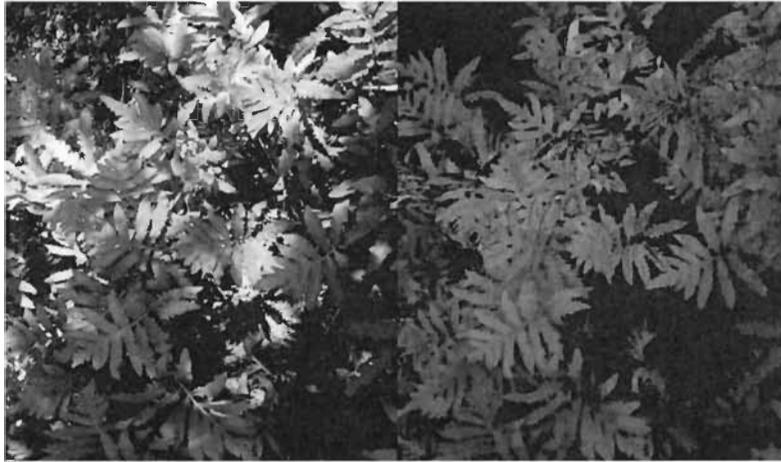


Figure 5 Application de l'indice ExG sur une image RGB

#### 2.4.2. L'indice d'excès de rouge ExR

Sur le même principe que l'indice d'excès de vert, mais inspiré du potentiel de l'œil humain à percevoir plus la couleur rouge que le vert ou le bleu (Meyer et al., 1999), l'ExR fait la discrimination de la couleur rouge, cette caractéristique est très utile en saison d'automne pour un grand nombre de végétaux qui prennent la couleur rouge. C'est également utile dans le cas d'une séparation de la végétation du sol si ce dernier a une couleur rouge. La Figure 6 permet de visualiser des plants de bleuets en automne dont les feuilles sont en processus de sénescence et rougissent puisque les pigments verts de chlorophylle font place aux pigments rouges d'anthocyanine. La formule de calcul du ExR est :

$$ExR = 1.3 - G$$



Figure 6 Application de l'indice ExR sur une image RGB

### 2.4.3. L'indice d'extraction de végétation CIVE

Cet indice a été proposé par (Kataoka, Kaneko, Okamoto, & S. Hata, 2003) dans leur recherche afin de discriminer les plantes du sol pour pouvoir étudier les étapes d'évolution de cette plante. La formule donnée pour cet indice est comme suit :

$$CIVE = 0.441R - 0.811G + 0.385B + 18.78745$$

Dans une étude comparative réalisée par (Hamuda, Glavin, & Jones, 2016), une liste de dix (10) indices de végétation sont étudiés avec des méthodes de seuillages afin de déterminer les performances de discriminations. Les chercheurs dressent un tableau de ces indices avec les avantages et les inconvénients de chacun. La comparaison de ces indices de végétation peut être obtenue par l'algorithme CIVE qui est le plus performant.

### 2.5. Les méthodes de seuillages

La performance de la méthode basée sur les indices de végétation peut être améliorée grâce au bon choix d'une méthode de seuillage, là encore il y a plusieurs techniques utilisées dans la littérature. Les méthodes de seuillage ont pour but de séparer deux classes grâce à un seuil prédéterminé ou à l'aide d'une méthode qui automatise le processus de sélection du seuil. Nous n'allons pas les détailler ici, mais nous nous contenterons d'exposer les plus utilisées.

#### 2.5.1. La méthode d'Otsu

Dans les méthodes vues précédemment, le seuil utilisé le plus souvent est soit le zéro ou alors l'utilisation de la méthode d'Otsu. Comme son nom l'indique, la méthode a été proposée par (Otsu, 1979) et permet d'automatiser le seuillage de façon non supervisée afin d'optimiser la séparation de deux classes dans une image en se basant sur l'histogramme des niveaux de gris.

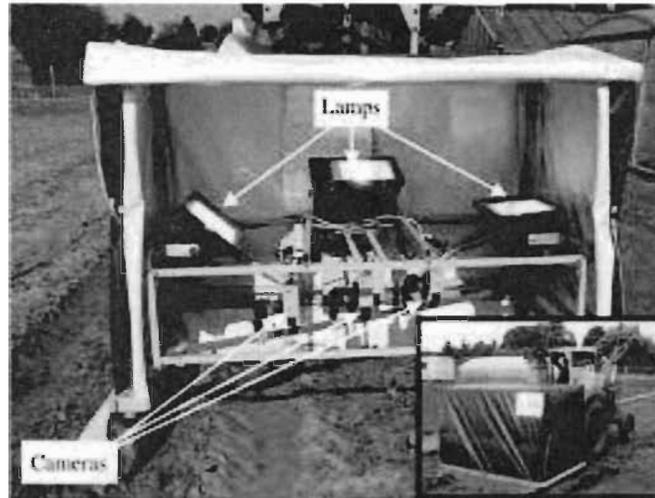


Figure 7 Le système de traitement d'image  
Source (Hemming & Rath, 2001)

Plusieurs autres méthodes de seuillages sont appliquées dans les travaux de recherches pour discriminer les plantes du sol ou différentes plantes. Si la méthode d'Otsu est la plus utilisée elle n'est cependant pas toujours le meilleur choix. Parfois, un simple seuil fixé à priori peut permettre de faire une bonne discrimination dans le cas où la lumière est contrôlée comme dans la recherche menée par (Hemming & Rath, 2001) ou les images ont été prises grâce à un dispositif avec un contrôle sur l'illumination (voir Figure 7). Le but étant de détecter les mauvaises herbes dans les champs de choux et de carottes.

Ainsi, on peut déduire que dépendamment du contexte de recherche ainsi que de l'approche choisie, les résultats donnés par telle ou telle méthode de seuillage peuvent varier. Parfois, là où il peut être pertinent d'utiliser une méthode automatisée donnant de meilleurs résultats. Dans d'autres situations un simple seuil fixé à priori produit des résultats satisfaisants.

## 2.6. La détection par apprentissage

Compte tenu des conditions parfois défavorables comme un environnement très ensoleillé ou sous un temps couvert l'approche basée sur les indices de végétations couleurs ne suffisent pas à discriminer efficacement les images voulues.

Des méthodes plus élaborées peuvent être utilisées en supplément ou en alternative afin de faire la classification des plantes comme les mauvaises herbes dans de tels environnements.

L'agriculture de précision de par la spécificité de ce domaine d'expertise requiert des connaissances humaines issues des spécialistes comme les agronomes, sont requises dans l'utilisation des algorithmes d'apprentissage machine. Ainsi l'utilisation de l'apprentissage supervisé ou semi-supervisé est plus adapté aux cas traités en agriculture de précision, même si

dans la littérature nous trouvons quelques recherches faites sur des cas avec l'apprentissage non supervisé souvent combiné avec d'autres méthodes d'analyse de données.

### 2.6.1. L'EASA (Environmentally Adaptive Segmentation Algorithm)

L'EASA est une méthode d'apprentissage semi-supervisé, dans laquelle certaines structures de données des images sont connues, il faut trouver les groupes dits « Naturel » de pixels dans une plage de couleurs correspondant aux objets de chaque classe. EASA est développé pour la discrimination de plantes dans un environnement à l'extérieur. La phase d'apprentissage dans le processus d'EASA réside dans l'étape de Clustering et l'algorithme schématisé dans l'organigramme (voir Figure 8) est constitué des étapes suivantes telles que décrites par (Tian & Slaughter, 1998) :

- Étape 1. Attribuez des points de départ prévus comme centre initial pour chaque groupe.
- Étape 2. Attribuez des points dans l'ensemble de données d'entraînement à leur point de départ le plus éloigné (en distance euclidienne).
- Étape 3. Recalculez les centres de groupe en fonction de tous les points qui leur sont attribués.
- Étape 4. Si la différence entre un ancien point de départ et le centre calculé du groupe qui le contient est supérieure à une valeur à priori établie, revenez à l'étape 2.
- Étape 5. Générez les résultats de clustering pour l'évaluation humaine et les affectations de classe d'objets.
- Étape 6. Si les résultats ne sont pas acceptables, générez de nouveaux points de départ et revenez à l'étape 1.
- Étape 7. Calculez des données statistiques pour chaque classe et créez une table de recherche basée sur un classificateur bayésien.

Contrairement aux méthodes statiques, cette approche donne un rendement satisfaisant en environnement moins illuminé, ainsi qu'un résultat meilleur en condition d'éclairage normal.

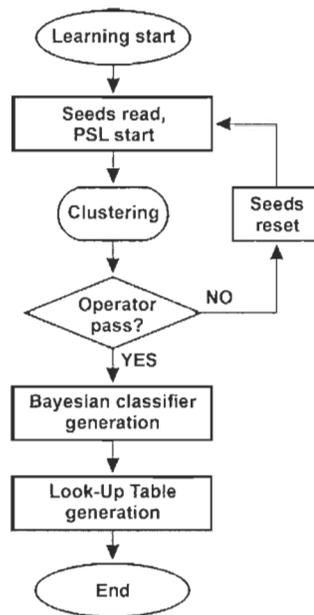


Figure 8 Organigramme de l'algorithme EASA, comprenant l'apprentissage, le calcul du classificateur et la génération de table de consultation.

Source (Ruiz-Ruiz, Gómez-Gil, & Navas-Gracia, 2009)

Par ailleurs, dans une autre recherche l'EASA a été modifié afin de tester différentes plages de couleurs autres que le RBV, entre autres le Hue-Saturation (HS) et le hue (H) ont été proposé par (Ruiz-Ruiz et al., 2009) afin d'optimiser la performance du processus en réduisant le temps de calcul.

## 2.6.2. SVM (Support Vector Machine)

Les machines à vecteurs de support sont considérées comme des classificateurs, cela veut dire qu'ils peuvent être utilisés afin de discriminer un groupe d'éléments d'un autre groupe. On peut donc voir leur grande utilité dans l'agriculture de précision pour discriminer les mauvaises herbes, ou les plantes atteintes de maladies ou les cultures d'autres éléments de leur environnement (plante et sol par exemple).

Les SVM sont des outils très puissants et peuvent être en apprentissage machine un excellent moyen dans des cas avec une petite ou moyenne base de données. On peut distinguer dans les SVM deux grandes branches : les SVM linéaires et les SVM non linéaires Marge maximale

Dans le cas d'un problème linéairement séparable, en plus du choix de l'hyperplan, son optimisation est un aspect important. Cette optimisation est possible avec la mise en place de marges qui maximisent la distance entre les éléments les plus proches des deux groupes séparés (voir Figure 9).

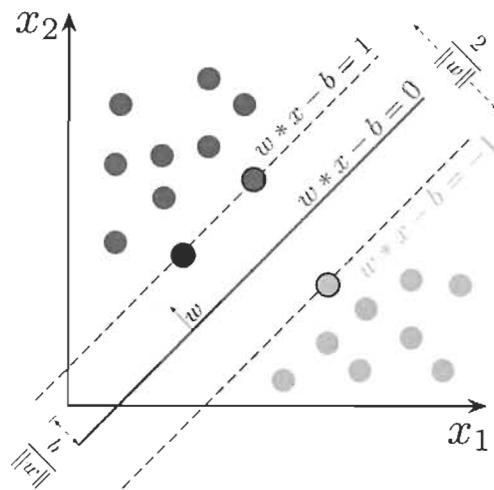


Figure 9 Illustration de séparation de deux classes avec SVM Linéaire à marge maximale.  
Source : (Wikipedia, 2020)

Dans certains cas il est impossible d'avoir un hyperplan séparant les classes de façon linéaire, ce problème peut être résolu en ajoutant une propriété comme dans l'exemple présenté dans la figure 10 (voir Figure 10).

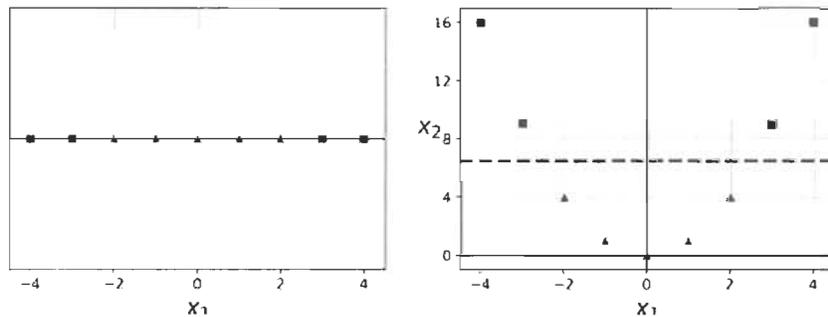


Figure 10 Ajout de propriété dans un SVM.  
Source : (Géron, 2019)

L'utilisation des SVM dans la classification est précédée par l'extraction des caractéristiques, cela est possible de différentes façons. Les plus fréquemment utilisées sont les méthodes statistiques et en particulier la GLCM (Gray Level Co-occurrence Matrix) (Haralick, Shanmugam, & Dinstein, 1973). La réduction du nombre de paramètres des textures peut être faite avec les approches d'analyse en composantes principales (ACP) afin de sélectionner les plus pertinentes d'entre elles. Une étude a été faite en appliquant ce protocole afin de discriminer les mauvaises herbes du blé. Les résultats de la détection ont donné 92.31 à 100% et pour la classification 100% contre 80% en utilisant un classificateur de réseaux de neurones Back propagation (Wu & Wen, 2009). Une étude similaire (Pulido et al., 2017), dans des conditions

extérieures, avec une extension du cas non linéaires et en utilisant une marge non-linéaire de forme Fonction Radiale de Base (RBF) et une réduction de 10 paramètres de textures avec une ACP a donné un résultat supérieur à 90% de classification (voir Figure 11).

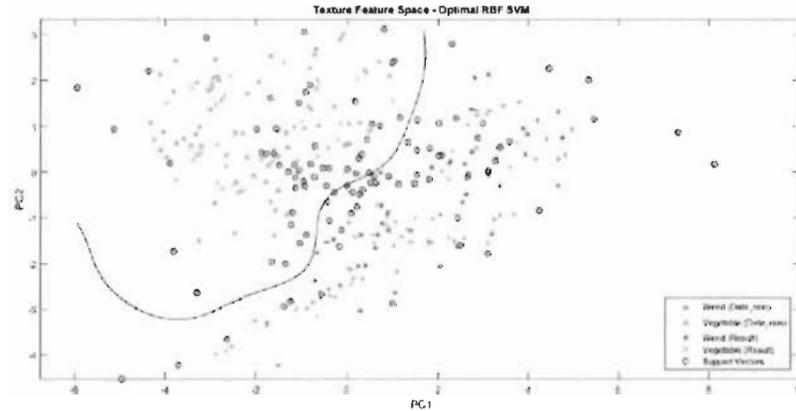


Figure 11 Classification SVM magenta : mauvaises herbes ; cyan : végétation.  
Source : (Pulido, Solaque et al. 2017)

### 2.6.3. Réseaux Bayésiens

Dans une approche probabiliste, les réseaux bayésiens offrent un bon moyen de classification des mauvaises herbes dans les champs agricoles. Ils sont souvent favorisés comme classificateurs dans les différentes recherches ((Ruiz-Ruiz et al., 2009), (Tian & Slaughter, 1998)).

Les classificateurs bayésiens naïfs se basent sur un modèle probabiliste. Ils réfèrent au théorème de Bayes formulé ainsi : prenant le cas d'une instance à classifier, représentée par un vecteur :

$$X = (x_1, \dots, x_n)$$

qui représente les variables indépendantes de n caractéristiques. Le modèle conditionnel est donné par :

$$p(C_k | x_1, \dots, x_n)$$

Pour chaque K possible il y a une probabilité  $p$  d'avoir la classe  $C_K$  connaissant  $x$ . Ce modèle est valable pour un petit jeu de données, mais devient inexploitable pour de larges jeux de données. Grâce au théorème de Bayes on peut le reformuler pour pouvoir mieux l'exploiter :

$$p(C_k | x) = \frac{p(C_k)p(x|C_k)}{p(x)}$$

Dans un cas de discrimination de mauvaises herbes dans des cultures en rangées, l'utilisation d'un classificateur bayésien est faite par (De Rainville et al., 2012) avec une procédure d'apprentissage non supervisée pour affiner la classification. La technique employée est de prendre les probabilités extraites d'une seule image, puis les appliquer au jeu de données non labélisé pour déterminer à quelles classes appartiennent ces données non labélisées (De Rainville et al., 2012).

#### 2.6.4. Les réseaux de neurones artificiels

Connaissant un regain d'intérêt très conséquent ces dernières années, les fondements des réseaux de neurones artificiels ne datent pourtant pas d'hier. En effet, la théorie fut introduite depuis 1943 par le neurophysiologiste Warren McCulloch et le mathématicien Walter Pitts (McCulloch, S., Pitts, & Walter, 1943).

S'en suivit des périodes de popularité et d'autres où ils sont moins priorisés, notamment dans les années 1990 où d'autres méthodes comme les SVM se sont montrées plus performantes et avec de meilleurs fondements mathématiques, les réseaux de neurones artificiels (ANN) étaient par conséquent moins utilisés.

Cela dit, nous vivons une période où les réseaux de neurones connaissent certainement le plus gros engouement depuis leur introduction et ce pour plusieurs raisons, dont nous pouvons citer quelques-unes :

- L'importante avancée au niveau du matériel informatique, notamment le développement de processeurs graphiques capable d'effectuer des lots de traitements en parallèle.
- La quantité de données disponibles et permettant d'alimenter les processus de traitement, avec l'utilisation massive d'internet et en particulier des services de visionnement et de chargement de vidéos, photos, ainsi que les services de traitement de textes.
- Le développement des algorithmes et des technologies faisant avancer la recherche dans le domaine des ANN. En effet, plus ils gagnent en popularité, plus de travaux sont réalisés utilisant ces ANN ce qui par conséquent permet d'explorer de nouveaux champs de recherche.
- Les bons résultats obtenus par les expérimentations récentes combler les limitations théoriques.

Tous ces points ainsi que d'autres font des réseaux de neurones artificiels un incontournable dans le domaine de traitement de données et en particulier la vision par ordinateur.

En réalité, les réseaux de neurones sont utilisés dans une branche de l'apprentissage machine appelée l'apprentissage profond, qui exploite le potentiel des réseaux de neurones artificiels en bénéficiant des derniers acquis technologiques mentionnés précédemment.

#### 2.6.4.1. Réseaux de neurones probabilistes (PNN)

Introduit en 1990 par (Specht & F., 1990), ils ont une architecture relativement simple, introduisant une partie probabiliste issue des réseaux bayésiens, l'architecture donnée par Specht est illustrée dans la Figure 12, et est constituée de quatre couches majeures, la couche d'entrée, la couche de modèle, la couche de sommation et la couche de sortie.

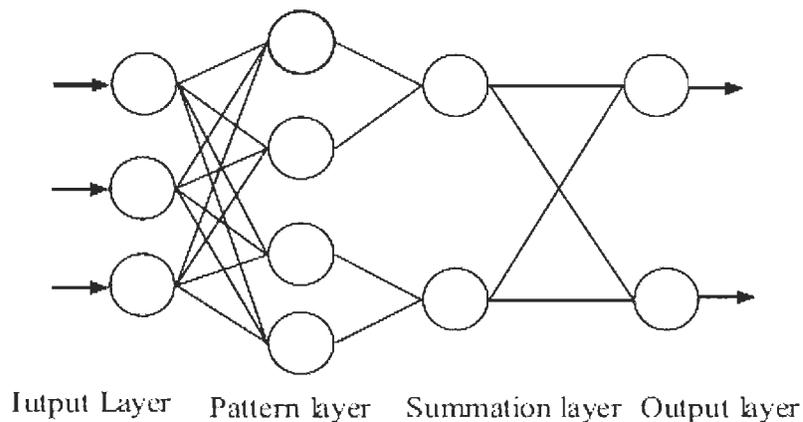


Figure 12 Réseaux de neurones probabilistes.  
Source : (Chen, Zhang, Su, & Guo, 2010)

La couche d'entrée transmet les entrées à la couche de modèle qui est totalement connectée avec elle, tandis que la connexion avec la couche de sommation est sélective selon la classe de modèle qu'elle représente. La couche de sortie est quant à elle pleinement connectée aux nœuds de la couche de sommation.

La couche d'entrée n'effectue pas de calculs, la couche de modèle assigne à chaque classe un neurone, puis la couche sommation effectue la somme des neurones cachés et transmet le résultat à la couche de sortie qui calcule la probabilité de l'appartenance à la classe.

L'étude comparative d'identification des mauvaises herbes dans une culture de maïs avec les PNN et la rétropropagation effectué par (Chen et al., 2010) donne avantage aux réseaux de neurones probabilistes avec un résultat de reconnaissance égale à 95%.

#### 2.6.4.2. Réseaux de neurones convolutif (CNN)

Les réseaux de neurones convolutif ou à convolution, sont utilisés particulièrement dans le traitement de données images, où chaque image est une matrice à deux dimensions de pixels.

L'utilisation des CNN nécessite une grande quantité de données et une bonne performance de calcul, ces deux particularités ont rendu possible la popularisation des CNN très récemment. En effet, depuis quelques années, on a une croissance énorme en termes de collecte de données notamment d'images ainsi qu'une amélioration considérable des technologies matérielles informatiques et en particulier les puces graphiques. Ces dernières sont capables d'effectuer des calculs parallèles très poussés.

Dans l'agriculture de précision, les CNN commencent à être exploités à des fins de détection et de classification, par exemple la classification des mauvaises herbes dans les champs de soja traité dans les travaux de (dos Santos Ferreira, Matte Freitas, Gonçalves da Silva, Pistori, & Theophilo Folhes, 2017). Dans une autre expérimentation, la recherche menée par (Tang et al., 2017), aboutit à la mise en place d'un modèle basé sur les paramètres d'apprentissage non supervisé du K-moyen combiné aux réseaux de neurones convolutifs.

Dans la suite de ce document et en particulier au troisième chapitre, nous aborderons plus en détail les CNN.

### 2.7. Conclusion

Nous avons eu un aperçu sur les différentes études ayant été faites sur le traitement de données, plus précisément sous forme d'images dans le domaine agricole. Dans la dernière section de ce chapitre nous avons introduit les réseaux de neurones. Nous n'avons pas abordé de façon détaillée les réseaux de neurones nous n'avons pas abordé les réseaux de neurones convolutifs, et pour cause, nous souhaitons leur dédier une partie entière dans le prochain chapitre, car cette approche est aujourd'hui la plus populaire et la plus promue. L'autre raison également est que c'est l'approche que nous avons privilégiée dans notre recherche.

Mais avant, nous allons décrire dans le chapitre trois, les données sur lesquelles nous avons fait notre travail, c'est-à-dire les mauvaises herbes, en abordant leurs espèces, les caractéristiques et les particularités de chaque espèce.

---

## **Chapitre 3. Cadre de recherche**

---

### 3.1. Introduction

Nous avons introduit dans le chapitre précédent les différentes techniques utilisées dans le traitement d'images et plus particulièrement la détection et la classification dans le domaine de l'agriculture de précision. Le présent chapitre sera dédié à la présentation de la démarche utilisée pour réaliser ce travail de recherche.

Les dernières années ont connu un regain énorme de l'utilisation des techniques de classification implémentées par réseaux de neurones et beaucoup de recherches sont faites dans le domaine de la vision artificielle. De plus, l'agriculture de précision n'échappe pas à la règle. Cela étant dit, il est toujours intéressant d'explorer de nouvelles solutions à notre problème consistant à détecter les mauvaises herbes.

Ce chapitre sera structuré de façon séquentielle comme le précédent cependant en suivant notre démarche et la méthodologie utilisées dans notre recherche.

### 3.2. Les mauvaises herbes

Il y a sans doute une multitude de mauvaises herbes dans la nature, certaines peuvent être très bénéfiques dans un contexte, mais considéré comme néfaste dans un autre, dans notre recherche nous nous sommes naturellement focalisés sur les plantes qui nuisent à la culture du bluet et donc considéré comme mauvaises herbes.

Nous nous sommes intéressés aux propriétés de ces mauvaises herbes autant qu'à celles qui sont pertinentes dans notre étude, c'est-à-dire leur aspect visuel, par conséquent les autres propriétés telles que l'odeur ou autres, ne seront pas considérées.

#### 3.2.1. La comptonie voyageuse

De son nom latin *Comptonia peregrina* ou en anglais Sweet Fern (douce fougère), cette espèce pousse de façon dense en grand nombre dans les terrains sablonneux ce qui crée de la concurrence pour les ressources avec les cultures du bluet.

La comptonie voyageuse se présente sous l'aspect d'un arbuste ayant une hauteur de 0,5 à 1,0 m. Ses feuilles ont un aspect particulièrement long et étroit, rapprochées avec de nombreux lobes. (Voir la Figure 13).



*Figure 13 La Comptonie voyageuse au printemps*

### 3.2.2. La kalmia à feuilles étroites

En latin *Kalmia angustifolia* on lui donne le surnom de Crevard de mouton à cause de sa toxicité envers certaines espèces animales, il est considéré comme la mauvaise herbe qui côtoie le plus le bluet dans son environnement naturel et par conséquent la plus répandue.

La plante a une hauteur de 60 à 80 cm avec un feuillage d'aspect elliptique et épais qui les rend tenaces et résistants aux conditions extrêmes de l'hiver (voir Figure 14). En automne, il prend une couleur rouge très distinctive (voir Figure 15).



*Figure 14 Kalmia feuille verte, image prise en été.*



*Figure 15 Kalmia feuille rouge, image prise en automne.*

### 3.2.3. Le quatre-temps

Également appelé Cornouiller du Canada de son nom latin *Cornus canadensis*. Le quatre-temps est répondu dans les forêts du Québec, il n'est pas présent dans tous les champs de bluet, mais quand il l'est, il se propage de façon étendue et ce qui le met en forte concurrence avec la culture.

Cette plante pousse près du sol, car sa longueur ne dépasse souvent pas les 20 cm (voir la Figure 16), d'une couleur verte vive et d'un rouge marron en automne (voir Figure 17), avec une forme à quatre feuilles, ce qui la rend très facile à distinguer.



Figure 16 Quatre-Temps vert.  
Source : ([repertoirequebecnature](#))



Figure 17 Quatre-Temps rouge.  
Source : ([repertoirequebecnature](#))

#### 3.2.4. Les touffes d'herbes

Surnommée Tussack ou Tussock en Anglais, il s'agit de Touffes d'herbe poussant à même le sol (voir Figure 18). Nous avons remarqué durant nos différentes missions pendant certaines périodes de l'année que ces plantes étaient répandues dans les champs de cultures.



Figure 18 touffes d'herbes prises lors d'une mission

### 3.3. L'apprentissage profond avec les réseaux de neurones convolutif (CNN)

Dans la section 6 du deuxième chapitre, nous avons évoqué les réseaux de neurones convolutif (CNN), leur récente domination dans le monde de traitement de données et de la vision artificielle notamment. Nous avons donné également des cas d'études mettant en pratique ce cas particulier des réseaux de neurones artificiels.

Dans ce qui suit, nous discuterons du choix des CNN pour le cas de la détection des mauvaises herbes dans les champs de bluets, suivi d'une brève introduction à leur origine. Par

la suite, nous allons donner une description de leur architecture suivie d'une liste non exhaustive de certaines d'entre elles et des recherches les ayant déjà appliquées.

### 3.3.1. Le choix des CNN pour notre cas d'étude

Dans notre recherche, nous voulons classifier les mauvaises herbes dans un champ de bluet sachant que nos données sont des images. En considérant les images avec leurs différents types (couleurs ou noirs et blancs) sont représentées au niveau abstrait par des matrices multidimensionnelles.

Dans une approche antérieure avec les réseaux de neurones le traitement de ce genre de données aurait été d'aplatir l'image en un seul vecteur avant de l'utiliser en entrée pour un réseau de neurones entièrement connecté. Dû à l'insensibilité de ces réseaux de neurones à l'ordre des propriétés, nous aurons des résultats similaires que ce soit en gardant les structures spatiales des pixels ou en permutant les colonnes de la matrice avant son introduction en paramètre pour l'apprentissage (Zhang, 2020).

De plus la disponibilité des ressources matérielles comme nous le verrons plus tard dans le document, ainsi qu'une banque d'images suffisante nous permettent d'adopter cette approche qui est devenue une référence dans la vision artificielle.

### 3.3.2. L'origine des CNN

À l'instar du domaine global que sont les réseaux de neurones artificiels, les réseaux de neurones à convolution se basent sur le modèle humain et animal, plus précisément le système oculaire. C'est d'abord (Fukushima, 1980) qui fut inspiré par les travaux effectués par les neurophysiologistes David Hunter Hubel et Torsten Nils Wiesel sur les structures du cortex visuel des animaux tels que les chats (Hubel & Wiesel, 1959), pour introduire la notion du «Neocognitron» (voir Figure 19). Il s'agit d'un modèle d'apprentissage automatique capable de reconnaître les formes géométriques similaires.

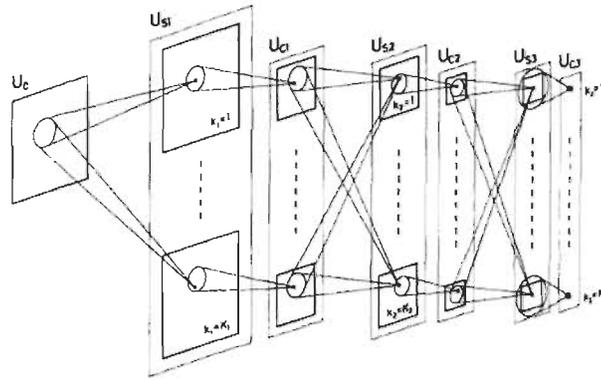


Figure 19 Interconnexions entre les couches dans le Neocognitron .  
Source : (Fukushima, 1980)

Une évolution graduelle au fur des années ayant aboutie à ce que nous connaissons actuellement comme les réseaux de neurones convolutifs. Une concrétisation du travail des chercheurs (Lecun, Bottou, Bengio, & Haffner, 1998) conclue par l'architecture LeNet-5 largement utilisée dans la reconnaissance des textes manuscrits.

### 3.3.3. Architecture des CNN

Comme tous les réseaux de neurones, les CNN sont composées de couches multiples, la première étant la couche d'entrée suivie d'un certain nombre de couches et puis par une couche de sortie qui donne le résultat de classification.

La structure typique est composée d'une succession de couches de convolutions, de couche Relu (Unité Linéaire Rectifiée) et couches de pooling, suivie d'une couche entièrement connectée et d'une couche de perte identifiée LOSS. La Figure 20 donne un aperçu de l'architecture d'un CNN typique.

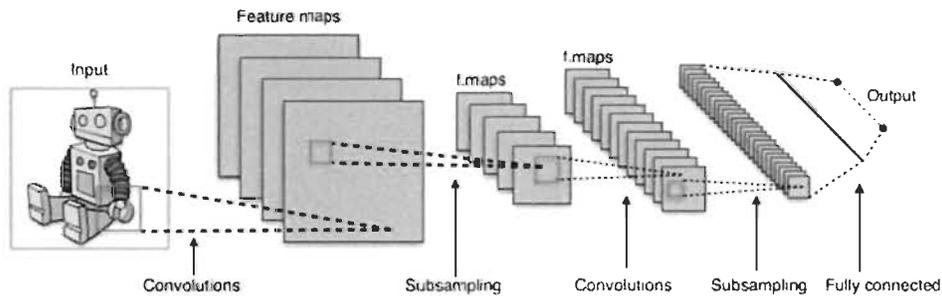


Figure 20 Architecture standard d'un réseau à convolutions.  
Source : (Wikipédia)

### 3.3.3.1. Les couches de convolution

La structure de base dans un réseau CNN est composée de couches convolutives, ces dernières sont connectées aux couches qui les précèdent de façon sélective. Ainsi, la première couche de convolution est connectée à la couche d'entrée qui est soit l'image dans le cas de la première couche de convolution ou alors les propriétés extraites de la couche précédente où chaque neurone est connecté au pixel qui lui correspond selon les paramètres comme illustré dans la Figure 21 .

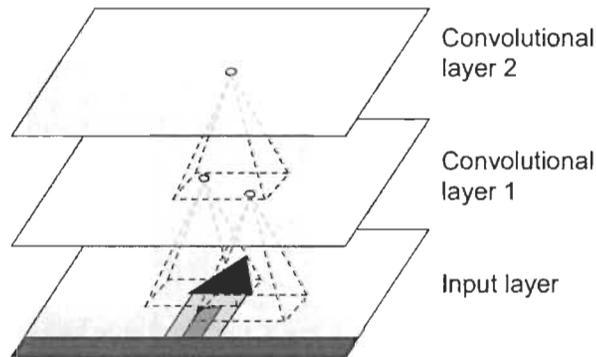


Figure 21 Couches de convolutions avec les récepteurs locaux en rectangle.  
Source (Géron, 2019)

### 3.3.3.2. Les couches de pooling

L'autre bloc composant un réseau de neurones convolutif est la couche de pooling. Elle fait le sous-échantillonnage, ce qui réduit considérablement le poids de calcul, l'utilisation de la mémoire ainsi que le nombre de paramètres.

Comme les couches de convolutions, les couches de pooling sont connectées à un certain nombre de neurones des sorties des couches précédentes. Quant à leurs fonctionnements, ils sont encore plus simples : ces couches n'ont pas de poids, car tout ce qu'elles font consiste à renvoyer le résultat d'une fonction d'agrégation soit maximal ou moyenne comme décrit sur la Figure 22.

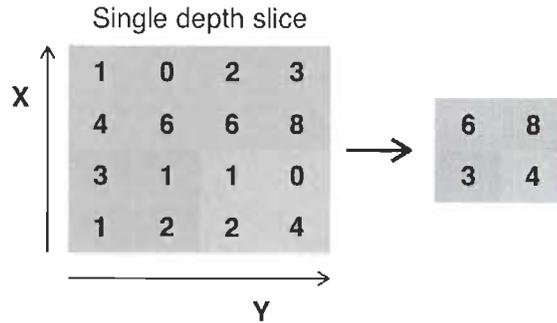


Figure 22 Max pooling avec un filtre  $2 \times 2$  et un pas de 2.  
Source : (Wikipédia)

Un autre avantage de l'utilisation des couches de pooling est l'élimination du surapprentissage. Cependant, ils ont aussi un inconvénient notamment la perte d'information due à la réduction. On peut minimiser cette perte en utilisant des petits filtres  $2 \times 2$  comme dans l'exemple illustré dans la Figure 22.

### 3.3.3.3. Les couches entièrement connectées

Tout comme l'indique leur nom, chacune de ces couches à une connexion totale avec la couche qui la précède. Nous pouvons utiliser avec la dernière couche entièrement connectée une fonction « sigmoïde » ou « softmax » pour la génération des prédictions des classes.

### 3.3.3.4. Les fonctions d'activations

La fonction d'activation peut être considérée comme une porte entre l'entrée et la sortie du neurone. Il s'agit d'une fonction mathématique qui détermine la sortie selon un seuil ou une règle.

L'une des fonctions d'activations la plus utilisée dans les réseaux de neurones, c'est la fonction ReLu (Rectified Linear Unit), la formule est donnée ainsi :

$$f(x) = \begin{cases} 0 & \text{si } x < 0 \\ x & \text{si } x > 0 \end{cases}$$

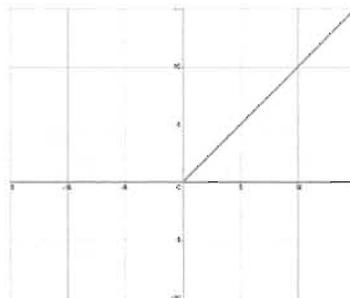


Figure 23 Graphe de la fonction ReLu

### 3.3.3.5. La fonction de perte (LOSS)

Dans le but d'optimiser l'apprentissage dans un réseau de neurones, nous devons minimiser les erreurs, c'est ainsi que nous pouvons définir le rôle de la fonction de perte, aussi appelée fonction de coût ou d'erreurs (Bengio, Goodfellow, & Courville, 2017).

Il y'a plusieurs fonctions pouvant être utilisées pour déterminer les erreurs, mais dans le cadre de l'apprentissage profond, les deux fonctions les plus utilisées sont :

- Le maximum de vraisemblance : Le maximum de vraisemblance cherche à trouver les valeurs optimales pour les paramètres en maximisant une fonction de vraisemblance dérivée des données d'apprentissage (Bishop, 1995).
- L'entropie croisée (Cross-Entropy) : Dans le cas des problèmes de classifications utilisant les réseaux de neurones convolutifs comme notre recherche, c'est la fonction d'entropie croisée qui est favorisée. Elle est donnée sous la formule suivante :

$$- \sum_{c=1}^M (y_c \log \hat{y}_c)$$

Où  $M$  est le nombre de classes, et  $\hat{y}_c$  est la prédiction du modèle pour cette classe  $\hat{y}_c$  a la valeur de 0 ou 1.

### 3.3.4. Exemples d'architectures de CNN

Il existe plusieurs architectures de réseaux de neurones convolutif, leur utilisation est relative au contexte. Ces architectures font souvent leurs preuves dans des défis d'apprentissage profond qui les rendent par la suite populaires et très utilisés. Voici notamment quelques-unes d'entre elles :

#### 3.3.4.1. LeNet-5

Initié par Yann LeCun (LeCun et al., 1989), pour arriver à la version finale conçue et souvent utilisée pour la reconnaissance des caractères manuscrits (Lecun et al., 1998).

#### 3.3.4.2. AlexNet

Le vainqueur du défi [ImageNet ILSVRC2012](#) avec un résultat de 15.3% en top-5 d'erreurs, devançant le second au classement de 10%. Le réseau est développé par Alex Krizhevsky d'où le nom ainsi que les chercheurs Ilya Sutskever et Geoffrey Hinton (Krizhevsky, Sutskever, & Hinton, 2017). Assez similaire à LeNet-5 avec plus de profondeur

et de largeur, cela est rendu possible notamment avec les avancées technologiques matérielles, en particulier, les processeurs graphiques et leur grand potentiel de calcul parallèle.

#### 3.3.4.3. GoogLeNet (Inception V1)

Cette architecture a remporté la version 2014 du défi ImageNet ILSVRC, elle est développée par (Szegedy et al., 2015), du département de recherche chez Google. Le résultat obtenu diminue considérablement la marge d'erreur comparée aux résultats obtenus dans les précédentes versions du concours ainsi qu'avec son second concurrent. Ceci en utilisant des sous-réseaux nommés « module Inception » qui sous-entend le fait d'aller plus profondément dans le réseau.

#### 3.3.4.4. VGGNet

Qui obtint la seconde place lors du ImageNet ILSVRC 2014, développé par (Simonyan & Zisserman, 2014), il a la particularité d'être simple avec deux ou trois couches de convolutions, puis une couche de pooling, s'ensuit une série similaire alternante de deux ou trois couches de convolutions et une de pooling jusqu'à avoir 16 couches de convolutions, et finir avec plusieurs couches complètement connectées successives.

#### 3.3.4.5. Xception

Qui signifie « Extrem Inception », c'est une variance de l'architecture GoogLeNet, ce modèle remplace les modules Inception par des modèles « séparable en profondeur ». Cette particularité permet notamment d'utiliser moins de paramètres et par conséquent moins de ressources ce qui lui vaut l'avantage d'être plus performant tout en étant moins gourmand en ressources matérielles.

#### 3.3.4.6. SENet

Ce CNN a remporté la compétition [ImageNet ILSVRC2017](#), cette architecture utilise des blocs nommés SE (Squeeze-AND-Excitation) qui sont une extension des précédentes architectures (Inception et ResNet). Ces blocs se concentrent sur la dimension de la profondeur des sorties auxquelles ils sont attachés, puis l'apprentissage se fait sur la base des propriétés qui sont les plus actifs ensembles (Hu, Shen, & Sun, 2017).

Dans la Figure 24 on peut voir la progression fulgurante des résultats durant ces dernières années, cela est notamment dû au fort engouement de la recherche pour le domaine

et en particulier pour les réseaux de neurones à convolutions, ainsi qu'aux avancées technologiques matérielles et le développement des GPU avec leur puissance de calcul.

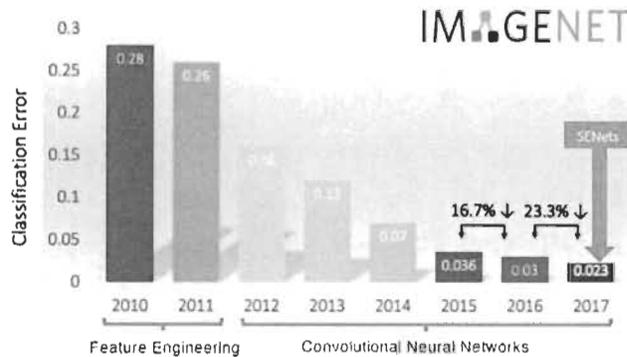


Figure 24 Progression des résultats des architectures CNN.  
Source : [image-net.org](http://image-net.org)

En plus de la classification, l'autre tâche tout aussi importante dans beaucoup de domaines incluant l'agriculture de précision, c'est la détection d'objets. En pratique la détection d'objets inclut deux aspects de l'apprentissage profond, la reconnaissance et la localisation. Voici d'autres architectures ayant performé dans cette tâche.

- R-CNN diminutif de Région et CNN est une technique développée par Ross Girshick (Girshick, Donahue, Darrell, & Malik, 2014). Le principe est d'avoir un modèle composé de 3 modules :
  - Module 1 : Générer et extraire les catégories indépendantes.
  - Module 2 : Extraire les propriétés de chaque catégorie.
  - Module 3 : Classer les propriétés par un classificateur (exemple SVM).

S'appuyant sur le succès du modèle R-CNN, le département de recherche de Microsoft se penche sur les limites de ce dernier ce qui amène Girshick à proposer une amélioration avec Fast R-CNN (Girshick, 2015). Puis une autre amélioration a été faite avec les travaux de Shaoqing Ren, et al. au département de recherche de Microsoft Faster R-CNN (Ren, He, Girshick, & Sun, 2015).

- YOLO diminutif anglais de « You Only Look Once » développé par Joseph Redmon (Joseph Redmon, Divvala, Girshick, & Farhadi, 2016), assurant une plus grande rapidité de détection dans l'optique d'une exécution en temps réel à une échelle de 45 frames par seconde. Une autre variante nommée Fast YOLO

fait encore plus vite avec 155 frames par seconde. L'approche de ce modèle consiste en l'utilisation d'un seul réseau de neurone pour prédire les cadres délimitant et les probabilités de classes directement d'une seule image. (Voir la Figure 25).

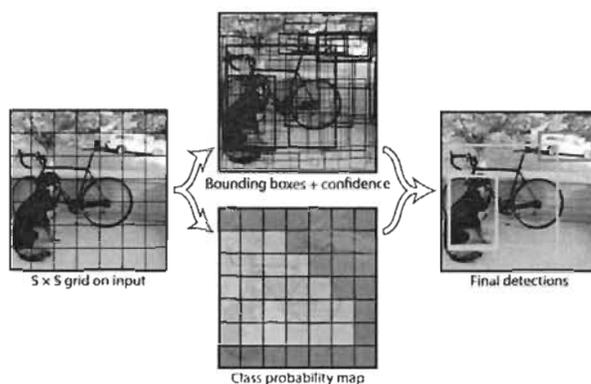


Figure 25 Illustration du modèle YOLO.  
Source : (Joseph Redmon et al., 2016)

L'architecture fut améliorée avec YOLOv2 où les chercheurs Joseph Redmon et Ali Farhadi dans leur article (J Redmon & Farhadi, 2016) ont été capables de prédire 9000 classes d'objets et ainsi donner au modèle le nom YOLO9000. Puis par la même équipe de chercheurs en 2018 la version YOLOv3 (Joseph Redmon & Farhadi, 2018).

## 3.4. Les outils de développement utilisés

### 3.4.1. Outils et technologies

Il existe plusieurs technologies pour réaliser l'apprentissage machine, mais toutes nécessitent une ressource matérielle relativement importante selon les données à traiter et le contexte de la recherche. Dans notre cas, les images constituent une base de traitement gourmande en ressource matérielle, ce qui implique le besoin d'une machine suffisamment puissante pour réaliser les expérimentations.

#### 3.4.1.1. Plateforme d'expérimentation

Après certaines expérimentations dans les systèmes locaux, que ce soit des ordinateurs personnels avec une configuration d'assez bonne qualité (un CPU Core i7 à 3.60 GHZ, 12 Go de mémoire et une carte graphique avec 1Go de mémoire dédiée), ou d'un ordinateur dont nous disposons au laboratoire LAMIA, notre choix s'est finalement tourné vers une utilisation plus

avantageuse en point de vue des ressources et de la disponibilité avec une des solutions dites dans les nuages.

Aujourd'hui plusieurs compagnies offrent un service d'utilisation de serveurs et de station de travail à distance, avec une allocation de ressource personnalisée selon les besoins. Les offres faites par ces compagnies sont soit payantes ou gratuites dépendamment des besoins.

Voici une liste non exhaustive des services d'apprentissage machine :

- Amazon Web Service ([AWS](#)).
- Microsoft [Azure](#).
- IBM [Watson](#).
- Salesforce [Einstein](#).

Pour notre recherche, nous avons privilégié un service web à la fois accessible et ouvert, offrant des solutions paramétrables selon les besoins.

#### 3.4.1.2. Google Colaboratory

Abréger en Google Colab, c'est le service d'apprentissage machine appartenant à la compagnie Google et plus précisément son département de recherche. Le service offre un accès libre et gratuit, mais limité à un environnement et selon les disponibilités et accessible à partir de n'importe quel ordinateur connecté à Internet. L'environnement est préalablement muni des ressources et bibliothèques nécessaires pour effectuer la recherche voulue.

#### 3.4.2. Le choix du langage de programmation

Pour nos expérimentations nous avons choisi le langage Python. Les raisons justifiant ce choix sont nombreuses. Le langage offre une grande flexibilité, une riche communauté offrant une source de documentation non négligeable et une parfaite intégration d'un large éventail des bibliothèques de recherches et de traitement d'images dont nous allons citer dans ce qui suit celle que nous avons utilisée.

#### 3.4.3. Bibliothèques et Framework

Il existe un grand nombre de bibliothèques et de Framework dans le domaine de l'apprentissage machine et l'imagerie par ordinateur, nous allons décrire quelques-unes qui nous avons utilisé dans notre travail.

##### 3.4.3.1. Keras et Tensorflow

Keras est une bibliothèque d'apprentissage machine, c'est l'une des plus utilisées. Écrite en Python, elle permet d'automatiser les tâches afin de faciliter les expérimentations dans le domaine de l'apprentissage profond. Keras s'intègre avec la plupart des autres bibliothèques d'apprentissage machine telle que Microsoft Cognitive Toolkit, R, Theano, ou PlaidML, mais elle est le plus souvent utilisée avec la bibliothèque Tensorflow. Cette combinaison très répandue à pousser la compagnie Google à l'intégrer directement à partir de la version 2.0 de Tensorflow.

### 3.4.3.2. Numpy

Étant donné que nous manipulons les images, nous traitons essentiellement des données représentées sur la machine sous forme de matrices, ainsi nous avons besoin d'algorithmes de traitement des matrices. C'est là, qu'intervient la bibliothèque Python Numpy qui regroupe un grand nombre de fonctions mathématiques capable de gérer des matrices de tailles très importantes (voir Figure 26).

```
>>> import numpy as np
>>> a = np.arange(15).reshape(3, 5)
>>> a
array([[ 0,  1,  2,  3,  4],
       [ 5,  6,  7,  8,  9],
       [10, 11, 12, 13, 14]])
>>> a.shape
(3, 5)
>>> a.ndim
2
>>> a.dtype.name
'int64'
>>> a.itemsize
8
>>> a.size
15
>>> type(a)
```

Figure 26 Un exemple d'utilisation de Numpy.  
Source : ([Numpy.org](http://Numpy.org))

### 3.4.3.3. Matplotlib

Pour pouvoir afficher et visualiser les résultats statistiques sous forme de courbes, histogrammes, points et autre, en 2D ou en 3D, nous faisons usage de Matplotlib qui est une interface de programmation applicable (API en Anglais), Matplotlib est une extension de Numpy pour le côté illustration mathématique (voir Figure 27).

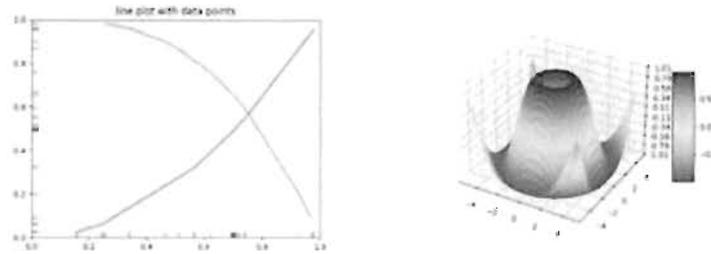


Figure 27 Illustration de schéma avec Matplotlib.  
Source : [Matplotlib.org](http://Matplotlib.org)

#### 3.4.3.4. OpenCV

Une des bibliothèques les plus utilisées dans le domaine de la vision par ordinateur, initialement développée par IBM en langage C et C++. Elle est également implémentée dans les langages Java, Javascript, Matlab et Python pour la manipulation des images.

OpenCV permet de faire une multitude de transformations dans les images et de vidéos telle que la détection d'objets, le seuillage d'images (comme la méthode d'Otsu), la segmentation, ainsi que la lecture et l'affichage des images.

#### 3.4.3.5. Tensorboard

C'est un outil de mesure et de visualisation que nous avons utilisé pour l'amélioration des résultats. Tensorboard est une API de la bibliothèque Tensorflow qui permet d'afficher les résultats des expérimentations effectuées grâce à cette dernière, en plus, Tensorboard fournit une multitude d'outils de comparaisons, d'évaluations et de suivi afin d'analyser et améliorer l'apprentissage des modèles.

### 3.5. Conclusion

Ce présent chapitre a été consacré aux contextes et technologies exploitées dans notre recherche. Nous avons d'abord présenté les mauvaises herbes qui constituent notre base de données. Ensuite, nous avons exposé l'architecture des réseaux de neurones à convolution qui sont la méthodologie de notre recherche. Pour finir, nous avons listé les différents outils qui nous ont permis d'exécuter nos expérimentations.

Dans le chapitre suivant nous détaillerons comment nous avons mis en pratique ces technologies et exploité ces données afin de faire nos expérimentations.

---

## **Chapitre 4. Méthodologie**

---

## 4.1. Plan de travail

Nous discuterons ici des démarches entreprises tout au long de notre travail, pour réaliser cela, nous avons élaboré un plan en plusieurs étapes schématisées dans la Figure 28.

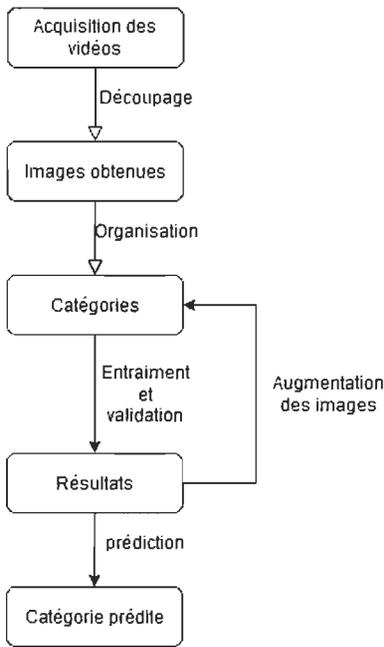


Figure 28 Les étapes de notre plan de travail

## 4.2. Acquisition des images

Parmi les méthodes d'acquisitions d'images précédemment citées, nous avons opté pour deux qui sont les plus accessibles. Il s'agit des drones ainsi que des captures d'images par un appareil photo en main, soit une caméra professionnelle ou un téléphone cellulaire muni d'une caméra de bonne qualité.

Pour notre recherche nous avons besoin d'un nombre assez important d'images exploitable. Afin de résoudre cette problématique nous avons utilisé une méthode de capture par vidéo, qui consiste à prendre des séquences vidéo de courtes durées (moins d'une minute) et ciblées pour ensuite extraire les images directement exploitables.

Nous avons effectué plusieurs missions à différentes périodes de l'année afin d'assurer une diversité des espèces de mauvaises herbes et leurs propriétés (stades d'évolutions et couleurs), comme nous pouvons le voir dans la Figure 29.



*Figure 29 Comptonie prise 25/10/2019*

Après que nous ayons pris les vidéos dans les champs de cultures, nous avons procédé à l'étape de création du corpus de données autrement dit des images, leurs organisations et aussi leurs labélisations.

#### 4.3. Organisation des données

Dans cette étape nous avons découpé les vidéos en images, pour se faire, il existe plusieurs méthodes plus ou moins automatisée. Nous avons opté pour l'utilisation d'un script Python. Lors des captures vidéo capturées, l'idée était de cibler des zones où proliférait qu'une seule espèce et par conséquent obtenir des séquences qui une fois découpée donnerait des images composées d'une espèce particulière de mauvaise herbe.

Mais cela ne fut malheureusement pas toujours le cas, ce qui nous à obliger par la suite à faire un traitement supplémentaire qui consistait à isoler les espèces dans les captures pour chaque prise afin d'obtenir un résultat suffisamment bon pour l'exploiter dans le traitement, comme nous pouvons le voir dans la Figure 30.

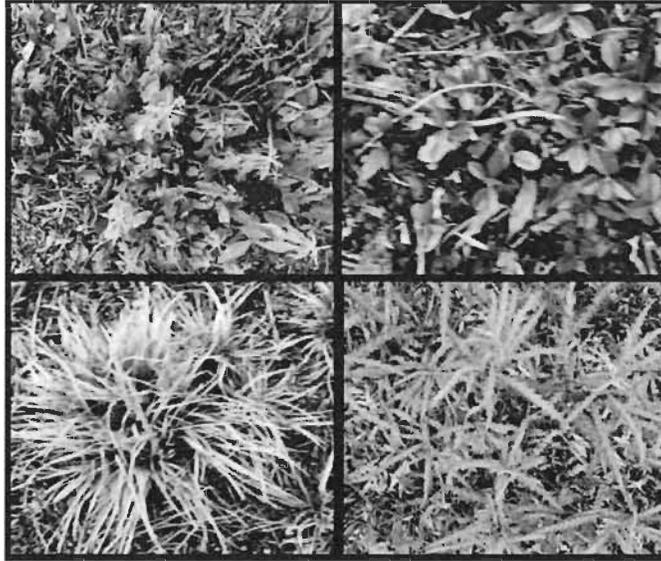


Figure 30 Exemple d'image de Kalmia, quatre-temps, touffe (tussock) et Comptonie.

Une fois les images regroupées par type de mauvaises herbes, il nous fallait construire une banque de données pour faire les différentes étapes de l'apprentissage. Pour ce faire, nous avons constitué nos dossiers selon l'organisation observé sur la Figure 31.

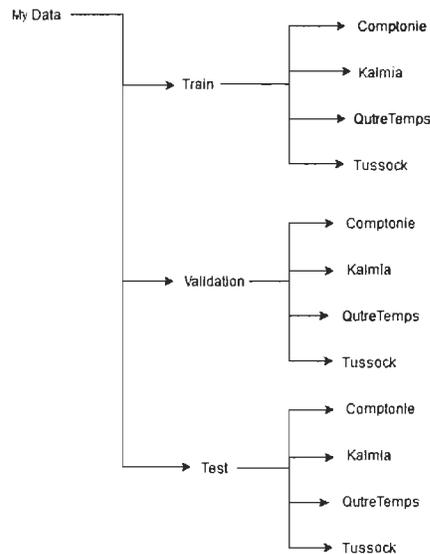


Figure 31 Organisation des données

Pour ne pas biaiser les résultats du traitement, nous avons tenu à ce que les images de chaque catégorie soient égales. La distribution exacte des données par catégorie et type est décrite dans le tableau 1. Le nombre d'images dans le dossier test n'a aucun impact sur

l'apprentissage, on a juste besoin de quelques images différentes afin de tester la prédiction une fois le traitement terminé.

	Entraînement	Validation	Test
Comptonie	1000	300	10
Kalmia	1000	300	10
QutreTemps	1000	300	10
Tussock	1000	300	10

Tableau 1 Nombre d'images dans le corpus de donnée

Les images contenues dans les dossiers d'entraînement représentent un peu plus de 70% alors que celles dans les dossiers de validations presque 30%.

#### 4.3.1. Le choix des images

Le choix du format et taille des images ainsi que le spectre de couleur utilisé est très important. En effet, cela impacte les résultats directement comme nous le verrons dans le prochain chapitre portant sur les résultats et la discussion. Des images trop petites risquent de diminuer les chances d'obtenir des résultats satisfaisants, cependant le choix d'images de grandes tailles entraîne forcément une durée de traitement plus longue et nécessite également beaucoup de ressources matérielles.

Quant au choix des spectres couleurs, il va impacter plus la qualité des résultats, par exemple une image en niveau de gris contient moins de propriétés qu'une image RBV. La simplicité et la réutilisabilité aussi interviennent dans ce choix, car si une image directement prise par un appareil donne de bons résultats, cela est avantageux.

Après plusieurs tests notre choix s'est porté sur des images en RVB avec une taille moyenne. Sachant que la taille des images prises est élevée (1932 x 2576 pixels), un redimensionnement s'impose alors afin d'optimiser la base de données pour le traitement. La taille choisie est (540 x 960 pixels).

#### 4.3.2. Entraînement et validation

Après l'acquisition et l'organisation des données, place à l'entraînement qui se compose de plusieurs parties successives.

##### 4.3.2.1. Le chemin d'accès aux données

Nous avons discuté au chapitre précédent de la plateforme utilisée pour les expérimentations, Google Colab. Vu ce choix, nous avons trouvé judicieux de charger les

données dans un dossier Drive afin de faciliter l'accès, afin de pouvoir exploiter les données une fois chargées sur Google Drive, il faut octroyer le droit d'accès du script aux dossiers contenant les données.

Une fois que le dossier est accessible, chacun des dossiers doit être déterminé distinctement, c'est-à-dire le dossier d'entraînement et celui de validation comme nous l'avons montré dans la section organisation des données.

Par la suite vient la partie de labélisation, là nous déterminons le nom de chaque classe, dans notre cas les quatre sortes de mauvaises herbes. Ensuite nous fixons les valeurs précédemment évoquées concernant la taille du lot de traitement, nombre d'itérations ainsi que la taille des images.

#### 4.3.3. Taille du lot de traitement

Malgré la réduction de la taille des images, le traitement nécessite une quantité énorme de ressource. Nous avons géré cette problématique en réduisant le nombre d'images traitées par lot, chaque lot prenant en charge 32 images à la fois.

#### 4.3.4. Nombre d'itérations

Le terme utilisé est *epoch*, c'est le nombre de fois où l'algorithme d'apprentissage est reproduit sur l'ensemble des données, chaque *epoch* donne un résultat et l'évolution de ces résultats constitue l'ensemble de l'apprentissage. Dans notre recherche nous avons pris une constante de 30 *epochs*.

Les itérations peuvent être interrompues avec un script si aucune amélioration n'est notée. Au terme des *epochs*, le modèle d'apprentissage est sauvegardé avec le meilleur résultat obtenu.

##### 4.3.4.1. Augmentation des données

Parmi les modules intégrés dans la bibliothèque Keras, nous pouvons utiliser le module ImageDataGenerator, cela nous permet entre autres de faire des manipulations et des augmentations sur les images en temps réel. Parmi ces manipulations nous utilisons le redimensionnement en divisant les valeurs de l'image par 255 afin de les restreindre dans le champ [0, 1], ainsi que le retournement horizontal. Nous pouvons également ajouter les rotations, les zooms et autres transformations préalablement implémentés dans le module ImageDataGenerator.

#### 4.3.4.2. Les sauvegardes et arrêts précoces

Au fur et à mesure que le traitement s'effectue, il se peut que notre modèle passe par des phases de non-apprentissage, cela veut dire qu'il arrive à une itération où il n'apprend plus. Pour y remédier nous implémentons des arrêts précoces avec la fonction `EarlyStopping` intégrée dans le module `Callback`. Ceci nous permet de sauter à l'étape suivante et ainsi prévenir une longue période de traitement inutile. Dans le même module, nous faisons appel à la fonction `ModelCheckpoint` pour faire la sauvegarde du modèle.

#### 4.3.5. Le modèle

Une fois que nous avons décrit dans le script les éléments du prétraitement des données, nous pouvons implémenter le modèle de l'apprentissage. Nous pouvons utiliser une des architectures précédemment décrites telles que VGGNet, Xception, SENet, cela peut par exemple être réalisé en appelant une fonction implémentée dans la bibliothèque Keras appelée VGG16. Sinon implémenter notre propre architecture. Nous avons opté pour la seconde option ce qui nous a permis entre autres de faire des modifications et ces expérimentations. L'architecture est illustrée par un schéma (voir Figure 32).

La fonction d'activation qui est utilisée pour toutes les couches sauf la dernière couche totalement connectée est la fonction « ReLu », pour la dernière couche c'est la fonction « SoftMax ».

Nous avons également utilisé un régularisateur l2 de 0.01 sur les couches de convolutions exceptées la première.

Le « dropout » est une technique qui permet d'éviter le surapprentissage (Srivastava, Hinton, Krizhevsky, Sutskever, & Salakhutdinov, 2014). Nous l'utilisons avant la dernière couche totalement connectée. Ce qui nous permettra en plus d'éviter le surapprentissage de gagner en performance et d'optimiser le temps des itérations.

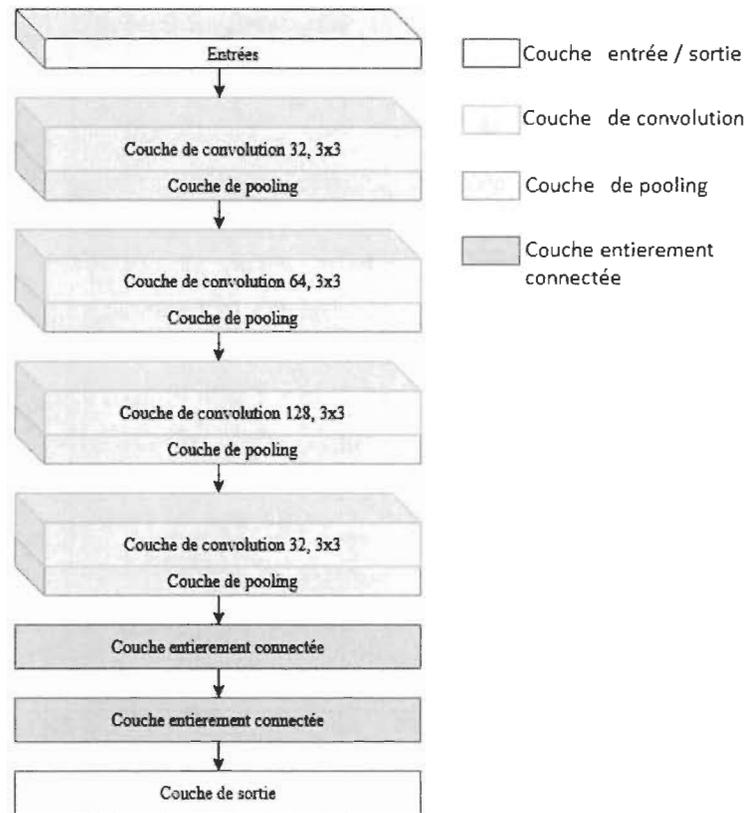


Figure 32 Diagramme du modèle avec les différentes couches le constituant.

Pour la compilation du modèle, nous avons utilisé l'optimiseur « Adam », la fonction de perte est « categorical\_crossentropy ». Il s'agit d'une variante de la fonction d'entropie croisée pour les problèmes non binaires comme le nôtre.

#### 4.3.5.1. La visualisation de la couche de convolution

Les CNN ont une structure un peu moins compacte que certains autres réseaux de neurones. Ils nous permettent alors d'avoir une vue plus interne de la structure ainsi que l'avancement des étapes d'apprentissage.

L'une de ces visualisations que nous pouvons voir, permet d'observer les caractéristiques extraites d'une entrée de la couche de convolution. Nous pouvons voir par exemple les caractéristiques de l'image d'entrée (voir la Figure 33).

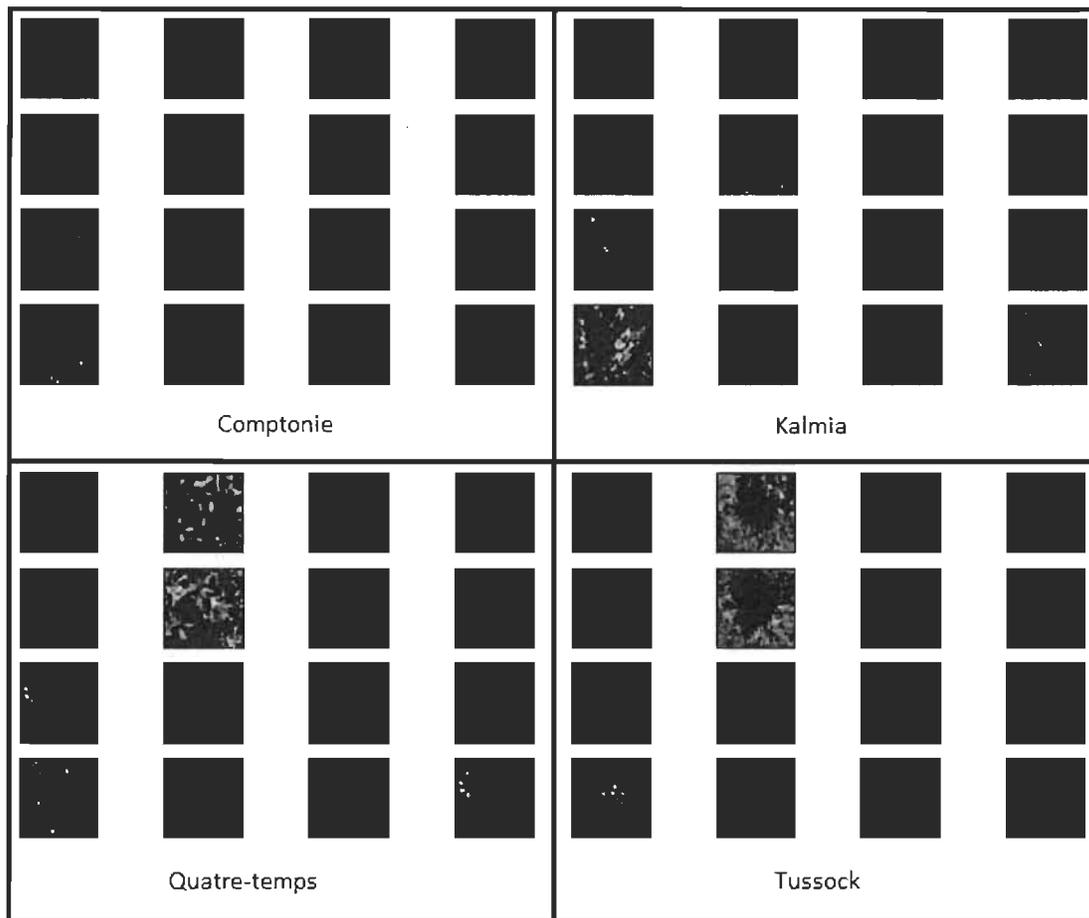


Figure 33 Caractéristiques extraites de la couche d'entrée des différentes espèces de mauvaises herbes

#### 4.3.6. Visualisation et amélioration

Pour pouvoir visualiser les résultats obtenus par notre modèle d'apprentissage et procéder aux réajustements permettant son amélioration, nous utilisons comme nous l'avons précédemment décrit le module « tensorboard » de la librairie TensorFlow.

#### 4.3.7. Évaluation des performances

Nous avons basé notre évaluation sur les principales métriques telles que : l'exactitude, la perte d'entropie croisée, la précision, le rappel, le score F1 et la matrice de confusion. Nous discuterons plus en détails de ces métriques dans le chapitre suivant consacré aux résultats et discussions.

Sachons que notre banque d'images est inédite, nous ne disposons pas d'éléments de comparaison antérieure, mais nos résultats peuvent être repris justement à des fins de comparaisons et d'amélioration comme nous allons le voir plus loin dans le document.

#### 4.4. Conclusion

Dans ce chapitre nous avons expliqué notre démarche, comment nous avons fait pour obtenir notre corpus de données, l'organisation de ces données et les traitements nécessaires pour qu'elles soient exploitables à des fins expérimentales. Ensuite, nous avons décrit comment nous avons mis en pratique des techniques d'apprentissage machine à l'aide des réseaux de neurones convolutifs appliquées à nos données.

Dans le prochain chapitre, nous allons présenter nos résultats, en faisant une mise au point sur les différents outils de mesures qui permettent de savoir à quel point le modèle est performant.

---

## **Chapitre 5.** Résultats et discussion

---

Après les multiples expérimentations menées durant notre recherche, les résultats que nous avons obtenus ont été différents. Les différences découlent des modèles utilisés, la forme et taille des images, le prétraitement effectué et toute autre manipulation ayant précédé l'apprentissage du réseau de neurones (voir la Figure 34).

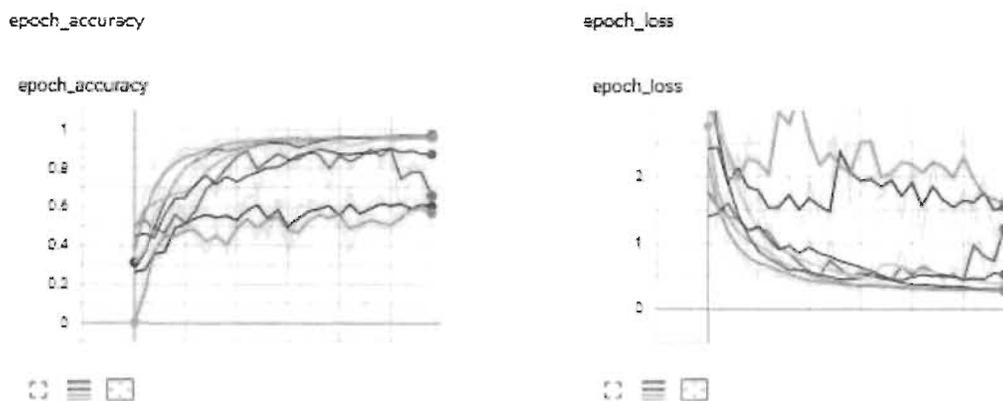


Figure 34 Résultats des réajustements du modèle

Le graphique sur la gauche représente le résultat de l'exactitude, celui de droite représente le résultat de perte. La figure 34 présente l'ensemble des résultats obtenus avec les entraînements que nous avons menés, où chaque couleur correspond à un entraînement distinct (avec la date et l'heure).

Dans ce qui suit, nous verrons les résultats obtenus et leur interprétation ainsi que les moyens qui nous ont permis de les améliorer. Ensuite, nous discuterons des potentielles améliorations et des perspectives qui peuvent être entreprises.

## 5.1. Les métriques de performances

### 5.1.1. L'exactitude de prédiction

Elle détermine à quel point les prédictions correspondent aux étiquettes. D'après la documentation de Tensorflow, cette métrique crée deux *variables* totales et *compteur* qui sont utilisées pour calculer la fréquence avec laquelle  $y\_pred$  la prédiction obtenue par le classificateur correspond à  $y\_vrai$  l'étiquette. Étant donné que nos données sont organisées de façon équilibrée entre les différentes classes, cette métrique est bien adaptée pour mesurer l'efficacité de la classification des prédictions.

Après plusieurs expérimentations nous avons pu atteindre une précision égale à 98% en apprentissage et 94.5% en validation (voir Figure 35).

La validation dépasse parfois la précision en apprentissage notamment au début d'apprentissage, cela est dû au *dropout* (défini dans la partie 3.5 du chapitre 4).

### 5.1.2. La perte d'entropie croisée catégorique

Nous avons discuté dans la partie méthodologique des fonctions de pertes ou de coûts, nous avons ajusté notre modèle en utilisant la fonction de perte d'entropie croisée catégorique. Les résultats optimaux nous ont permis d'avoir un minimum égal à 0.298 en apprentissage et 0.323 en validation (voir Figure 35).

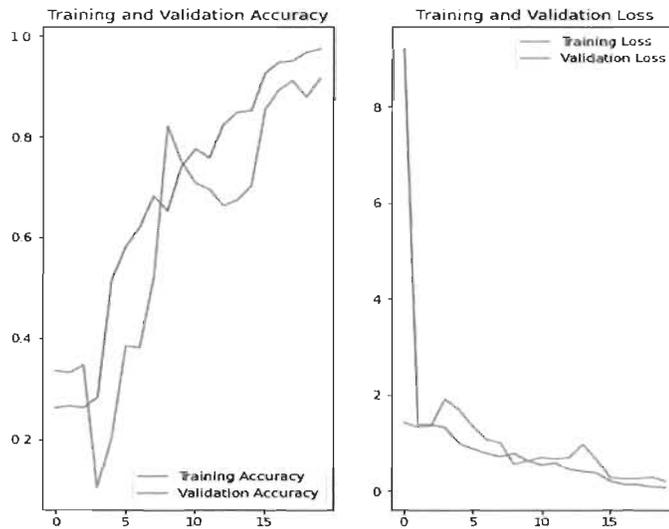


Figure 35 L'exactitude (à gauche) et la perte (à droite) pour l'entraînement et la validation.

Sur le schéma à droite dans la figure 35, nous pouvons voir que la valeur de la fonction de perte LOSS a une tendance descendante ce qui signifie que les résultats s'améliorent. La valeur que nous avons obtenue à la fin des itérations est très concluante.

Après plusieurs réajustements nous avons pu obtenir ces résultats au bout de 20 passages (epochs), un nombre de passages trop grand peut induire le modèle à faire du surapprentissage qui va dégrader les résultats, comme nous pouvons le voir dans la Figure 36, au-delà de 30 passages le résultat d'apprentissage régresse soudainement.

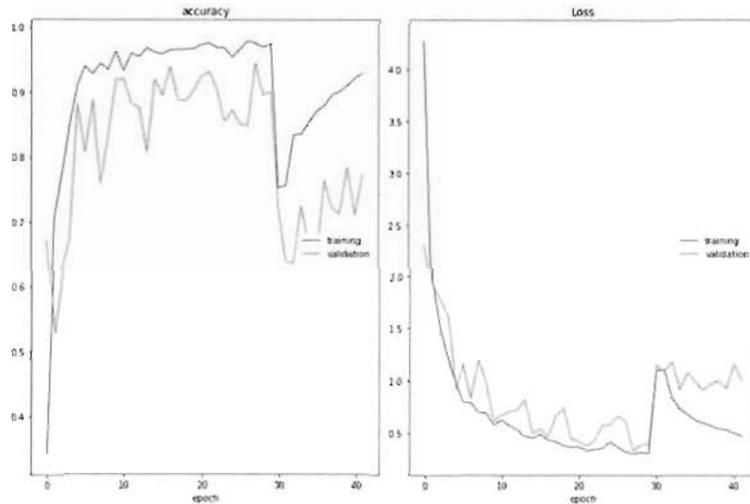


Figure 36 entraînement d'un modèle avec 50 passages

### 5.1.3. La précision et le rappel et F1

La précision détermine la qualité de la classification des prédictions vraies faites par le modèle, elle est donnée par la formule suivante :

$$\text{précision} = \frac{\text{vraies positives}}{(\text{vraies positives} + \text{faux positives})}$$

Le rappel décrit l'exactitude de la prédiction des classes positives dans l'ensemble des prédictions pertinentes. Différemment de la précision, il prend en considération les faux négatifs. Il est donné par la formule suivante :

$$\text{rappel} = \frac{\text{vraies positives}}{(\text{vraies positives} + \text{faux négatives})}$$

La mesure F1 est la moyenne des deux métriques précédentes, la mesure F1 est très bien adaptée pour les cas de données non équilibrées, F1 peut être calculé ainsi :

$$F1 = 2 \frac{(\text{précision} \times \text{rappel})}{(\text{précision} + \text{rappel})}$$

Le Tableau 2 représente le rapport de classification que nous avons obtenu avec le modèle optimal.

	Précision	Rappel	F1	Support
Comptonie	0.94	0.99	0.97	300
Kalmia	0.98	0.92	0.95	300
QuatreTemps	0.98	1.00	0.99	300
Tussock	0.91	0.90	0.91	300

Tableau 2 Rapport de classification

Les résultats que nous avons obtenus, concordent bien avec ceux de l'exactitude et sont tous supérieurs à 90%. Les meilleures prédictions que nous avons obtenues avec le modèle, sont celles de la classe quatre temps, ce résultat est en particulier dû à la distinction des images dans la base d'entraînement et pour la validation de cette classe caractérisée par leur teinte rougeâtre ce qui donne une grande valeur ajoutée au classificateur avec des paramètres supplémentaires. Cette distinction ne serait pas possible dans le cas d'utilisation d'image de niveaux de gris par exemple.

Les images de la classe Comptonie sont aussi bien classifiées, nous interprétons cela par la densité de cette plante dans les images, car parmi les quatre classes, la Comptonie est celle qui présente le plus grand nombre de plantes dans les images.

#### 5.1.4. La matrice de confusion

La matrice de confusion nous permet de voir exactement ce que notre modèle donne comme résultat de classification.

Le résultat de la matrice de confusion est donné dans le Tableau 3, la classe quatre temps qui a donné le meilleur pourcentage de classification comme nous l'avons décrit avec le rapport de classification présenté au Tableau 2 donnant la précision, le rappel et F1. Toutes les prédictions que retournent le classificateur pour les images de ces classes sont des vraies positives. Pour la classe *comptonie*, deux images sont mal classées. La classe *tussock* parmi les 300 prédictions. La classe *kalmia* a 24 images classées à tort dans la classe *tussock*, cette dernière a les prédictions les moins justes, avec 270 biens classés sur les 300.

	Comptonie	Kalmia	QuatreTemps	Tussock
Comptonie	298	0	0	2
Kalmia	0	276	0	24
QuatreTemps	0	0	300	0
Tussock	19	6	5	270

Tableau 3 Matrice de confusion

## 5.2. Discussion

Nous avons pu observer au travers des différentes expérimentations obtenues que les résultats sont vraiment très variés et les possibilités d'optimisation nombreuses. Après la détermination d'un modèle pouvant servir d'une bonne base pour l'apprentissage, nous avons pu faire des améliorations jusqu'à arriver à un résultat satisfaisant avec les contraintes que nous avions.

Les contraintes dans notre recherche et comme toujours dans les problèmes de classification sont relatives aux données traitées. En effet l'étape la plus importante avant l'entraînement des données est la préparation et le prétraitement des données.

Le nombre limité d'images que nous possédions, ainsi que leurs qualités étaient relativement restreintes pour un problème de classification, cela est dû entre autres à la simplicité des moyens d'acquisitions.

Nous avons également rencontré des problèmes de classification à cause des images qui contenait parfois d'autres classes, par exemple une image de kalmia contenant des touffes d'herbes, cela induit le classificateur en erreur et crée un biais dans les résultats.

Nous jugeons donc qu'il est primordial d'accorder à la phase préparation et prétraitement le temps nécessaire pour construire une bonne base de données qui soit bien exploitable pour n'importe quel problème d'apprentissage.

Les réajustements et améliorations sont également utiles pour obtenir une performance optimale.

L'augmentation des données dans la phase d'apprentissage est utile sous certaines conditions, dans notre cas par exemple cela ne nous a pas permis d'améliorer les résultats.

## 5.3. Conclusion

Les moyens techniques et les outils pour les problèmes de classification sont de plus en plus nombreux, le choix de ces derniers peut se faire selon le cas traité et les données que nous

possédons. Mais il est avant tout primordial de bien préparer les données, en particulier quand il s'agit d'un problème de classification d'images en couleurs et/ou les couleurs font partie des propriétés indispensables pour la classification.

Il faut donc prendre le temps de collecter un nombre conséquent d'images des différentes espèces que l'on souhaite classer, choisir les résolutions suffisamment grandes pour extraire les propriétés tout en les minimisant pour optimiser le temps d'apprentissage afin de pouvoir faire plusieurs ajustements et arriver à un bon résultat de classification.

Dans notre recherche, malgré les limitations que nous avons rencontrées, nous avons réussi à obtenir de bons résultats (98% en apprentissage et 94.5% en validation).

En effet, le manque d'une base de données au départ nous a contraints à construire la nôtre en effectuant les missions de capture d'images sur de longues périodes, d'organiser et traiter ces images afin d'avoir une base exploitable pour les expérimentations.

---

## **Chapitre 6.** Conclusion et perspectives

---

Afin d'optimiser la production dans les cultures de bluet, il est indispensable de limiter la prolifération des mauvaises herbes dans les champs. Que ce traitement soit fait de manière conventionnelle ou non, il est primordial de classer les différentes espèces afin d'optimiser leur éradication.

Le principal défi auquel nous devons faire face, c'était de pouvoir discriminer les différentes espèces de mauvaises herbes qui se répandent dans les champs de bluets. Pour y parvenir, nous devons avoir une base de données composées d'images des différentes espèces. Ensuite, passer ces images sous plusieurs étapes de traitements, d'organisations et par la suite, faire les expérimentations nécessaires.

Nous avons utilisé des moyens simples pour la capture de séquences vidéo et d'images, que nous avons transformées au bout de plusieurs étapes en une base de données contenant les différentes catégories de mauvaises herbes.

Par la suite, nous avons basé notre recherche sur une approche récente et efficace. Ainsi, nous avons utilisé les réseaux de neurones à convolution pour classer les quatre principales espèces qui affectent les cultures de bluet.

Les moyens techniques que nous avons utilisés dans notre traitement, sont des moyens disponibles et accessibles n'importe où il y'a un accès à Internet. Les technologies telles que les langages de programmation et les bibliothèques que nous avons utilisées sont libres d'accès et gratuites. Même si le traitement avec les réseaux de neurones à convolution nécessite une configuration avancée, les services dans les nuages comme Google Collab la plateforme que nous avons utilisée rendent la tâche plus accessible.

Avec le modèle de réseaux de neurones à convolution que nous avons présenté, basé sur d'autres modèles ayant prouvé leur efficacité dans d'autres champs de recherches. Nous avons par la suite fait des ajustements afin d'améliorer ce modèle et par conséquent améliorer nos résultats.

Nous avons démontré qu'il est possible de déterminer avec une bonne précision les catégories de ces mauvaises herbes. Les résultats que nous avons obtenus sont concluants et démontrent l'efficacité de notre démarche.

Les résultats prometteurs que nous avons obtenus ouvrent des perspectives pour d'autres recherches. En effet, ils peuvent servir de bases pour d'autres projets. Avec la possibilité d'utiliser la base de données déjà constituée pour faire d'autres expérimentations et utiliser les résultats que nous avons comme base de comparaison, ou alors en construisant d'autres bases de données avec plus d'images.

L'étude peut également être reprise dans un système de détection et de traitement de mauvaises herbes, en implémentant par exemple la partie que nous avons traitée, c'est-à-dire la classification dans un système intelligent de détection et d'éradication des mauvaises herbes.

## Bibliographie

- Ballesteros, R., Ortega, J. F., Hernández, D., & Moreno, M. A. (2014). Applications of georeferenced high-resolution images obtained with unmanned aerial vehicles. Part I: Description of image acquisition and processing. *Precision Agriculture*, 15(6), 579-592. doi:10.1007/s11119-014-9355-8
- Bengio, Y., Goodfellow, I., & Courville, A. (2017). *Deep learning* (Vol. 1): MIT press Massachusetts, USA:.
- Bishop, C. M. (1995). *Neural networks for pattern recognition*: Oxford university press.
- Campbell, J. B., & Wynne, R. H. (2011). *Introduction to Remote Sensing, Fifth Edition* (5 ed.): Guilford Press. ISBN : 9781609181772
- Chen, L., Zhang, J., Su, H., & Guo, W. (2010). Weed identification method based on probabilistic neural network in the corn seedlings field. *International Conference on Machine Learning and Cybernetics*. doi:10.1109/ICMLC.2010.5580822
- Chung, S., Breshears, L. E., & Yoon, J.-Y. (2018). Smartphone near infrared monitoring of plant stress. *Computers and Electronics in Agriculture*, 154, 93-98. doi:10.1016/j.compag.2018.08.046
- De Rainville, F.-M., Durand, A., Fortin, F.-A., Tanguy, K., Maldague, X., Panneton, B., & Simard, M.-J. (2012). Bayesian classification and unsupervised learning for isolating weeds in row crops. *Pattern Analysis and Applications*, 17(2), 401-414. doi:10.1007/s10044-012-0307-5
- dos Santos Ferreira, A., Matte Freitas, D., Gonçalves da Silva, G., Pistori, H., & Theophilo Folhes, M. (2017). Weed detection in soybean crops using ConvNets. *Computers and Electronics in Agriculture*, 143, 314-324. doi:10.1016/j.compag.2017.10.027
- Ducournau, S., Feutry, A., Plainchault, P., Revollon, P., Vigouroux, B., & Wagner, M. H. (2004). An image acquisition system for automated monitoring of the germination rate of sunflower seeds. *Computers and Electronics in Agriculture*, 44(3), 189-202. doi:10.1016/j.compag.2004.04.005
- Fukushima, K. (1980). Neocognitron: a self organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biol Cybern*, 36(4), 193-202. doi:10.1007/BF00344251
- Géron, A. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow* (2nd Edition ed.): O'Reilly Media, Inc. ISBN : 9781492032649
- Girshick, R. (2015). *Fast r-cnn*. Paper presented at the Proceedings of the IEEE international conference on computer vision.
- Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). *Rich feature hierarchies for accurate object detection and semantic segmentation*. Paper presented at the Proceedings of the IEEE conference on computer vision and pattern recognition.

- Hamuda, E., Glavin, M., & Jones, E. (2016). A survey of image processing techniques for plant extraction and segmentation in the field. *Computers and Electronics in Agriculture*, 125, 184-199. doi:10.1016/j.compag.2016.04.024
- Haralick, R. M., Shanmugam, K., & Dinstein, I. (1973). Textural Features for Image Classification. *IEEE Trans Syst Man Cybern, SMC-3*, 610-621. doi:10.1109/TSMC.1973.4309314.
- Hemming, J., & Rath, T. (2001). PA—Precision Agriculture. *Journal of Agricultural Engineering Research*, 78(3), 233-243. doi:10.1006/jaer.2000.0639
- Hu, J., Shen, L., & Sun, G. (2017). Squeeze-and-Excitation Networks. *CoRR*, abs/1709.01507. Retrieved from <http://arxiv.org/abs/1709.01507>
- Hubel, D. H., & Wiesel, T. N. (1959). Receptive fields of single neurones in the cat's striate cortex. *J Physiol*, 148, 574-591. doi:10.1113/jphysiol.1959.sp006308
- Kataoka, T., Kaneko, T., Okamoto, H., & S. Hata. (2003). Crop growth estimation system using machine vision. *IEEE/ASME International Conference on Advanced Intelligent Mechatronics, vol.2*, pp. b1079-b1083. doi:10.1109/aim.2003.1225492
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6), 84-90. doi:10.1145/3065386
- LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., & Jackel, L. D. (1989). Backpropagation Applied to Handwritten Zip Code Recognition. *Neural Computation*, 1(4), 541-551. doi:10.1162/neco.1989.1.4.541
- Lecun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278-2324. doi:10.1109/5.726791
- Luoto, M., Toivonen, T., & Heikkinen, R. K. (2002). Prediction of total and rare plant species richness in agricultural landscapes from satellite images and topographic data. *Landscape Ecology*, 17(3), 195-217. doi:10.1023/A:1020288509837
- McCulloch, S., W., Pitts, & Walter. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5(4), 115-133. doi:10.1007/bf02478259
- Meyer, G., Hindman, T., & Laksmi, K. (1999). *Machine vision detection parameters for plant species identification* (Vol. 3543): SPIE.
- Otsu, N. (1979). A Threshold Selection Method from Gray-Level Histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1), 62-66. doi:10.1109/tsmc.1979.4310076
- Pulido, C., Solaque, L., & Velasco, N. (2017). Weed recognition by SVM texture feature classification in outdoor vegetable crop images. *Ingeniería e Investigación*, 37. doi:<http://dx.doi.org/10.15446/ing.investig.v37n1.54703>

- Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). *You only look once: Unified, real-time object detection*. Paper presented at the Proceedings of the IEEE conference on computer vision and pattern recognition.
- Redmon, J., & Farhadi, A. (2016). YOLO9000: Better, faster, stronger. arXiv 2016. *arXiv preprint arXiv:1612.08242*.
- Redmon, J., & Farhadi, A. (2018). Yolov3: An incremental improvement. arXiv 2018. *arXiv preprint arXiv:1804.02767*, 1-6.
- Ren, S., He, K., Girshick, R., & Sun, J. (2015). *Faster r-cnn: Towards real-time object detection with region proposal networks*. Paper presented at the Advances in neural information processing systems.
- Ruiz-Ruiz, G., Gómez-Gil, J., & Navas-Gracia, L. M. (2009). Testing different color spaces based on hue for the environmentally adaptive segmentation algorithm (EASA). *Computers and Electronics in Agriculture*, 68(1), 88-96. doi:10.1016/j.compag.2009.04.009
- Simonyan, K., & Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition.
- Specht, & F., D. (1990). Probabilistic neural networks. *Neural Networks*, 3(1), 109-118. doi:10.1016/0893-6080(90)90049-q
- Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, 15, 1929-1958. Retrieved from <http://jmlr.org/papers/v15/srivastava14a.html>
- Szegedy, C., Wei, L., Yangqing, J., Sermanet, P., Reed, S., Anguelov, D., . . . Rabinovich, A. (2015). *Going deeper with convolutions*. Paper presented at the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- Tang, J., Wang, D., Zhang, Z., He, L., Xin, J., & Xu, Y. (2017). Weed identification based on K-means feature learning combined with convolutional neural network. *Computers and Electronics in Agriculture*, 135, 63-70. doi:10.1016/j.compag.2017.01.001
- Tian, L. F., & Slaughter, D. C. (1998). Environmentally adaptive segmentation algorithm for outdoor image segmentation. *Computers and Electronics in Agriculture*, 21(3), 153-168. doi:10.1016/s0168-1699(98)00037-4
- Wikipedia. (2020). Support vector machine. Retrieved from [https://en.wikipedia.org/w/index.php?title=Support\\_vector\\_machine&oldid=956661830](https://en.wikipedia.org/w/index.php?title=Support_vector_machine&oldid=956661830)
- Wu, L., & Wen, Y. (2009). Weed/corn seedling recognition by support vector machine using texture features. *African Journal of Agricultural Research*, 4, 840-846.