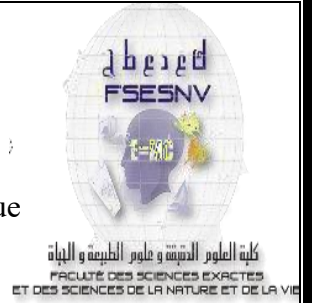




République Algérienne Démocratique et populaire
Ministère de l'enseignement supérieur et de la recherche scientifique
Université Larbi Tébessi - Tébessa



Faculté des Sciences Exactes et des Sciences de la Nature et de la Vie

Mémoire de fin d'étude

Pour l'obtention du diplôme de MASTER

Domaine : Mathématiques et informatique

Filière : Informatique

Option : Réseaux et sécurité informatique

Thème

**Réseaux des capteurs sans fil intelligent pour la
détection des mauvaises herbes dans les
applications agricoles**

Réalisé par :

Lakehal Djihene

Devant le jury :

Mr. Gharbi Abdelhakim	MAA	Université Larbi Tébessi	Président
Mr. Tag Samir	MAA	Université Larbi Tébessi	Examineur
Dr. Merzoug Soltan	MCB	Université Larbi Tébessi	Encadreur

Date de soutenance :13 juin 2022

REMERCIEMENT

Tout d'abord, je remercie Dieu le Tout-Puissant de m'avoir donné la force morale et physique et de m'avoir permis d'accomplir ce travail.

*Je tiens à remercier mon encadrant **Dr. Merzoug Soltan**, maître de conférences classe B à l'université de Tébessa, pour m'avoir dirigé dans ce travail. Je le remercie pour sa*

Disponibilité, son suivi, ses précieux conseils et son aide.

*Je remercie chaleureusement Monsieur le Président du jury, **Mr Gharbi Abdelhakim**, et à Monsieur examinateur le **Mr Tag Samir**, de m'avoir honoré en participant au jury de mon mémoire et pour revoir mon travail.*

Merci à mon père, à ma mère, à mes sœurs, pour leur compréhension, leur patience et leur soutien. Alors, à toute ma famille.

Ils ont toujours été la source de mon succès, pour cela je ne saurai jamais comment les remercier.

Je prie Dieu de me les garder le plus longtemps possible.

Dédicace

A mes chers parents,

Que nulle dédicace ne puisse exprimer ce que je leurs dois, pour leur bienveillance, leur affection et leur soutien... Trésors de bonté, de générosité et de tendresse, en témoignage de mon profond amour et ma grande reconnaissance « Que Dieu vous garde ».

A mes chères sœurs

(Khouloud, omnia, nour el imen)

En témoignage de mes sincères reconnaissances pour les efforts qu'ils ont consenti pour l'accomplissement de mes études. Je leur dédie ce modeste travail en témoignage de mon grand amour et ma gratitude infinie

A tous mes amis

Pour leur aide et leur soutien moral durant l'élaboration du travail de fin d'études.

A tous ma Famille

A tous ceux dont l'oubli du nom n'est guère celui du cœur...

« De l'union « si » avec « mais » naquit enfant nommé « jamais » »

« Il n'y a pas de « si » ni de « mais », il faut réussir »

Djihene

Résumé

Les réseaux de capteurs sans fil sont aujourd'hui utilisés dans de nombreux domaines tels que le militaire, l'agriculture, etc. En agriculture, des capteurs sans fil qui enregistrent les données d'une caméra que vous embarquez dans ces capteurs et les transmettent sous forme numérique à un ordinateur pour traiter la détection des mauvaises herbes existantes

Dans ce contexte, nos recherches visent à faciliter la détection des adventices les plus courantes dans la région de Tébessa. Grâce à nos expériences, nous avons développé deux modèles de détection basés sur le réseau de neurones Convolutif (masque R-CNN, yolov5), capables de détecter avec une bonne précision et de bons résultats.

Mots-clés : capteurs, sans fils, convolutions, réseaux de neurones, apprentissage profond, apprentissage automatique, mauvaises herbes.

Abstract

Wireless sensor networks are used today in many fields such as military, agriculture, etc. In agriculture, wireless sensors that record data from a camera that takes it on board these sensors and transmits it in digital form to a computer for processing the detection of existing weeds.

In this context, our research aims to facilitate the detection of the most common weeds in the Tebessa region. Our experiments, we have developed two detection models based on a convolutional neural network (Mask R-CNN and YOLO) that are able to accurately detect and provide good results.

Keywords: Wireless sensor network, convolutions, neural networks, Deep Learning, weeds.

المخلص

تستخدم شبكات الاستشعار اللاسلكية اليوم في العديد من المجالات مثل العسكرية، الزراعة، وما إلى ذلك. في الزراعة، أجهزة الاستشعار اللاسلكية التي تسجل البيانات من الكاميرا التي تأخذها على متن هذه المستشعرات وتنقلها في شكل رقمي إلى جهاز كمبيوتر لمعالجة اكتشاف الأعشاب الموجودة.

في هذا السياق، يهدف بحثنا إلى تسهيل الكشف عن الحشائش الأكثر شيوعًا في منطقة تبسة. بفضل تجاربنا، قمنا بتطوير نموذجين للكشف على أساس الشبكة العصبية التلافيفية (قناع و الولوج) القادرة على الكشف بدقة وتقديم نتائج جيدة

الكلمات المفتاحية: التلافيف، استشعار، لاسلكية، الشبكات العصبية، التعلم العميق، التعلم الآلي، الأعشاب الضارة.

Table des matières

Liste des figures	10
Liste des tableaux	12
Liste des abréviations	13
Introduction générale	14
Introduction générale	1
Chapitre 01.....	3
Généralité sur les réseaux des capteurs sans fils	3
1 Introduction	4
2 Définition	4
3 La caractéristique des réseaux des capteurs	5
4 Architecture physique d'un nœud capteur.....	6
4.1 Unité d'acquisition	6
4.2 Unité de traitement	6
4.3 Unité de communication.....	7
4.4 Batterie.....	7
5 Problèmes d'énergie dans les Réseaux de capteurs sans fil	7
5.1 Les causes de gaspillage d'énergie	7
6 Les domaines d'application des réseaux de capteurs	8
7 Domaine agricole (agriculture de précision)	9
8 Conclusion	9
Chapitre 02.....	10
Généralité sur les mauvaises herbes et la zone d'étude	10
1 Introduction	10
2 C'est quoi une mauvaise herbe.....	10
3 Source des mauvaises herbes	10
4 Avantages et inconvénients des mauvaises herbes	12
4.1 Avantage :	12
4.2 Inconvénients	12
5 Les types des mauvaises herbes existant dans Tébessa	12
6 Situation géographique de la région d'étude	14
7 Caractéristiques de la région	15
7.1 Relief	15

7.2	Climat.....	15
7.3	Agricole.....	16
8	Conclusion.....	16
	Chapitre 03.....	17
	État de l'art	17
1	Introduction	18
2	L'acquisition d'image	18
2.1	Satellite.....	18
2.2	Les drones.....	19
2.3	Caméra.....	19
2.4	L'utilisation des téléphones intelligents	20
3	Notion d'Intelligence artificielle	20
4	L'apprentissage automatique (Machine Learning).....	20
4.1	L'apprentissage supervisé	21
4.2	L'apprentissage non supervisé.....	21
4.3	L'apprentissage par renforcement	22
4.4	L'apprentissage profond	22
5	Apprentissage Profond (Deep Learning).....	22
5.1	Définition et origine	22
5.2	Les réseaux de neurones.....	23
5.3	Quelque type d'algorithmes utilisés en Deep Learning.....	24
5.3.1	Les réseaux de neurones profonds (DNN).....	24
5.3.2	neurones récurrents (RNN)	24
5.3.3	Les réseaux récurrents à mémoire court et long terme (LSTM)	24
5.3.4	Les réseaux antagonistes génératifs (GAN)	24
5.3.5	Les réseaux de neurones Convolutif (CNN).....	25
6	Le Réseau Neuronal Convolutif (CNN).....	25
6.1	Définition	25
6.2	Les couches de CNN	25
6.2.1	Couche Convulsive	25
6.2.2	Couche de correction ReLU	26
6.2.3	Couche de pooling.....	26
6.2.4	Couche entièrement connectée (fully-connected)	27
7	Réseaux de neurones de convolution pour la détection d'objets.....	27
7.1	R-CNN.....	27

7.2	Yolo	28
7.3	Faster R-CNN.....	28
7.4	Mask R-CNN	29
8	Les méthodes utilisées pour détecter les mauvaises herbes	29
8.1	Synthèses.....	32
9	Conclusion	33
	Chapitre 04.....	35
	Contribution.....	35
1	Introduction	36
2	Les mauvaises herbes	36
3	L'acquisition des images	37
4	Conception globale	37
4.1	collection de donnée	39
4.2	prétraitement de donnée	39
4.2.1	Division de donnée.....	40
4.2.2	Limitier les mauvaises herbes.....	40
4.3	Exportation des données	41
4.4	Importation de modèle	41
4.4.1	Mask R-CNN 2017.....	41
4.4.2	Modèle YOLO V5(2020) :.....	43
4.5	L'entraînement de modèle	44
4.6	Evaluation de modèle.....	44
4.7	Prédiction et test des nouvelles images	44
5	Conclusion	45
	Chapitre 05.....	35
1	Introduction	47
2	Les outils de développement utilisés	47
2.1	Language de programmation.....	47
3.	Le matériel.....	49
4.	Simulation	50
5.	Réalisation	50
5.1	Mask R-CNN	50
5.1.1	Connexion avec Colab.....	50
5.1.2	Importer les bibliothèques et le modèle MaskRCNN	51
5.1.3	Accès aux données	53

5.1.4	L'entraînement du modèle	54
5.1.5	Validation de la base de données	54
5.1.6	Prédiction	55
5.2	Yolov5 :	56
5.2.1	Importer la bibliothèque et le modèle	56
5.2.2	Accédé aux données	56
5.2.3	L'entraînement du modèle	57
5.2.4	Validation.....	57
6	Résultats et discussion	58
6.1	Taux d'erreur (loss)	58
6.2	Rapport d'évaluation de detection	61
7	Comparaison	63
8	Test de nouvelle image	64
9	Discussion	66
10	Conclusion	67
	Conclusion général et perspectives	70
	Conclusion générale.....	71
	Perspectives	72
	Listes des Référence.....	71

Liste des figures

Figure 1: Réseaux de capteurs sans fils	4
Figure 2: Architecture d'un système basée sur RCSF.....	5
Figure 3: Architecture physique d'un nœud capteur.....	6
Figure 4: Domaines d'applications de RCFS.....	8
Figure 5: Les sources des mauvaises herbes	11
Figure 6: Localisation géographique de la wilaya de Tébessa	14
Figure 7: Exemple d'un système d'acquisition d'image satellite	19
Figure 8: La relation entre l'IA, apprentissage automatique et profond ainsi que réseau de neurone ...	23
Figure 9: Exemple d'un réseau de neurones artificiel	23
Figure 10: Architecture d'un réseau de neurones Convolutif	25
Figure 11: Schéma du parcours du résultat de convolution	26
Figure 12: Rectified Linear Unit (ReLU).....	26
Figure 13: La couche max pooling.....	27
Figure 14: Exemple d' Architecture R-CNN	27
Figure 15: Architecture Faster R-CNN	28
Figure 16: L'architecture de modèle MaskRCNN.....	29
Figure 17: Les mauvaises herbes étudiées (extrait de la base de données)	37
Figure 18: Conception de notre système basé sur RCSF	38
Figure 19: La conception global du notre projet	39
Figure 20: Echantillon de donnée.....	40
Figure 21: Limitation des mauvaises herbes	41
Figure 22: L'architecture de detection les mauvaises herbes basée sur le Mask R-CNN.....	42
Figure 23: L'architecture FPN.....	42
Figure 24: Architecture de YOLOV5.....	43
Figure 25: Connexion avec Colab.....	50
Figure 26: Import des bibliographies	51
Figure 27: Configuration de l'entraînement des mauvaises herbes.....	51
Figure 28: La configuration de modèle	52
Figure 29: L'affichage de configuration total des modèles utilisés	52
Figure 30: Schéma de base de données	53
Figure 31: Unzip de donnée	53
Figure 32: Accès aux données.....	53
Figure 33: La fonction du l'entrainement	54
Figure 34: L'exécution de train	54
Figure 35: Le contenu de validation.....	54
Figure 36: La fonction de prédiction.....	55
Figure 37: Prédiction images de validation.....	55
Figure 38: Les bibliothèques importer	56
Figure 39: Accès aux données.....	56
Figure 40: L'entrainement du modèle yolov5	57
Figure 41: Validation des images.....	57
Figure 42: Le taux d'erreur (loss) de modèle Mask RCNN du 40 epochs.....	58
Figure 43: Le taux d'erreur de class.....	59
Figure 44: Le taux d'erreur de cadre	59
Figure 45: Le taux d'erreur de masque	59
Figure 46: Le taux d'erreur de class de la région	60

Figure 47: Le taux d'erreur de cadre de la région	60
Figure 48: Les résultats obtenus à partir du modèle YoloV5	61
Figure 49: L'évaluation de detection mAP avec Mask RCNN	62
Figure 50: L'évaluation de précision de Mask R-CNN	62
Figure 51: L'évaluation de detection avec YOLO	63
Figure 52: Test des nouvelles images	66

Liste des tableaux

Tableau 1: Classement des mauvaises herbes suivant leur période.	13
Tableau 2: les travaux de recherche rapportés par des chercheurs sur la détection des adventices	32
Tableau 3: Tableau de paramètre de simulation.....	50
Tableau 4: tableaux de comparaison	63

Liste des abréviations

RCFS : Réseaux des Capteurs Sans Fils

WSN : Wireless Sensors Network

IA : intelligence artificiel

DL : Deep Learning (apprentissage profond)

ML : Machine Learning (apprentissage automatique)

CNN : Convolutional Neural Network

R-CNN: Region Based Convolutional Neural Networks

DNN: Deep Neural Networks

RNN: Recurrent Neural Networks

LSTM : Long Short-Term Memory

GAN : Les réseaux antagonistes génératifs

ReLU: Rectified Linear Unit

RVB: Rouge, Vert, Bleu

Via: VGG Image Annotator

JSON: JavaScript Object Notation

ResNet: Residual Network

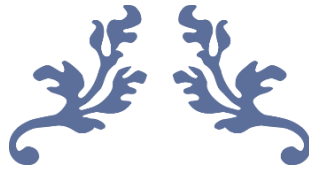
FPN: Feature Pyramide Network

Yolo: You Only look once

PANet: Path Aggregation Network

CSP: Cross-Phase Partial Networks

GPU: Graphics Processing Unit



Introduction générale



Introduction générale

L'Organisation des Nations Unies pour l'alimentation et l'agriculture prévoit que nous devons augmenter la production alimentaire mondiale de 70 % au cours des prochaines décennies pour nourrir la population projetée en 2050. Environ 2 50 000 espèces végétales ont été observées et parmi elles, il y a 250 espèces associées aux mauvaises herbes. Celles-ci ont un impact significatif sur la réduction de la production agricole.

Le traitement des données (apprentissage automatique ou apprentissage profond) est en plein essor, touchant de nombreux domaines et disciplines du monde de la recherche ou professionnel. Le secteur agricole n'échappe pas à cette réalité, notamment avec l'agriculture de précision intelligente.

La disponibilité et la facilité d'obtenir de plus en plus de données à traiter, notamment des images rendues faciles grâce à la technologie des nouveaux moyens de captation tels que les caméras, les drones, et surtout les smartphones, a grandement contribué au développement des moyens qui lui permettent de être exploité. Diverses méthodes et techniques ont été explorées afin de traiter les images capturées afin de pouvoir extraire des données utilisables pour améliorer le débit quantitatif et qualitatif.

Tandis que les capteurs sans fil qui enregistrent les données d'une caméra que vous intégrez dans ces capteurs et les transmettent sous forme numérique à un ordinateur pour traiter la détection des mauvaises herbes existantes.

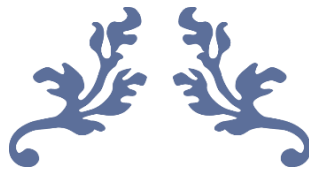
Nos recherches porteront sur l'un des problèmes majeurs de l'agriculture, la propagation des mauvaises herbes sur les terres agricoles. Une détection précoce et mieux ciblée des mauvaises herbes pourrait jouer un rôle majeur dans la promotion de l'agriculture intelligente, car cela permettrait une réduction significative de l'utilisation d'herbicides et de produits chimiques dans le cas d'une agriculture conventionnelle, manuelle ou mécanisée.

Cette mémoire est composée de cinq chapitres :

Dans le premier chapitre, nous avons introduit les différents concepts liés aux réseaux des capteurs sans fils et le deuxième chapitre est intéresser par la région géographique de Tébessa et déterminé ses caractéristiques agronomiques et climatiques, ainsi que nous avons abordé l'identification des herbes en définissant leur définition, leur origine, les types présents dans la zone étudiée et leurs avantages.

Ensuite troisième chapitre est consacré aux méthodes et techniques de détection de mauvaises herbes basées sur l'apprentissage profond, à étudier le concept d'intelligence artificielle et dans les détails de l'apprentissage profond (Deep Learning) et à révéler un type d'algorithme à utiliser. L'apprentissage profond et ses différents types et principes car ces dernières années ont vu un énorme regain d'utilisation des techniques de détection mises en œuvre par les réseaux de neurones.

Puis le quatrième chapitre sera consacré à la présentation de la conception général du notre système et à la présentation de l'environnement de développement. Enfin, le dernier chapitre sera celui qui est appliqué et la simulation avec une discussion des résultats obtenus.



Chapitre 01

Généralité sur les réseaux des capteurs sans fils



1 Introduction

Un réseau de capteurs sans fil (plus connu sous le nom de réseau de capteurs sans fil (WSN) en anglais) est constitué d'un ensemble de terminaux ou nœuds appelés Capteurs qui communiquent sur des liaisons radio sans infrastructure fixe préalable. Le réseau doit fonctionner de manière autonome sans intervention humaine.

Ce type de réseau est constitué de centaines ou de milliers d'éléments (capteurs) dont le but est de collecter des données de l'environnement, de les traiter et de les transmettre au monde extérieur. Ces capteurs sont utilisés dans plusieurs domaines d'application: surveillance de l'environnement, opérations militaires et de sécurité, médecine, etc.

2 Définition

Un réseau de capteurs sans fil (RCSF ou WSN : Wireless Sensor Network) est constitué d'un grand nombre de nœuds répartis dans une region donnée, qui mesurent des grandeurs physiques ou surveillent des événements et, si nécessaire, envoient Informations collectées à un ou plusieurs facteurs de collecte à l'aide d'une connexion sans fil. Dans un tel réseau, chaque nœud est un dispositif électronique doté de capacités de calcul, de stockage, de communique et d'énergie [1].

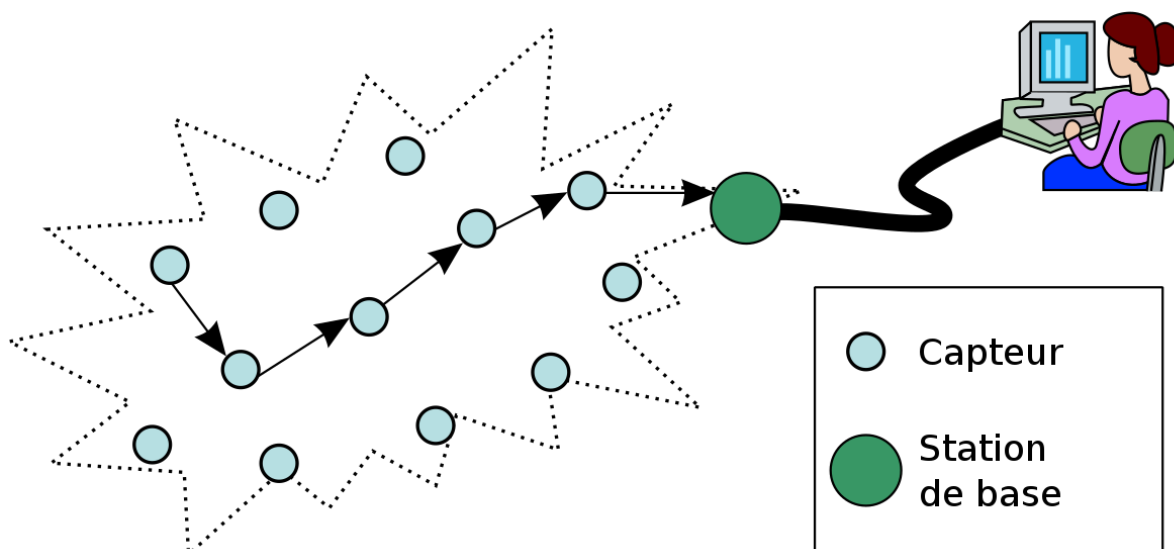


Figure 1: Réseaux de capteurs sans fils [1]

Ces nœuds capteurs sont organisés en champs « sensor fields » (voir figure suivante).

Chacun de ces nœuds a la capacité de collecter des données et de les transférer à la passerelle (dit "sink" en anglais ou puits) par l'intermédiaire d'une architecture multi-sauts. Le puits transmet ensuite ces données par Internet à l'ordinateur central «Gestionnaire » pour analyser ces données et prendre des décisions [2].

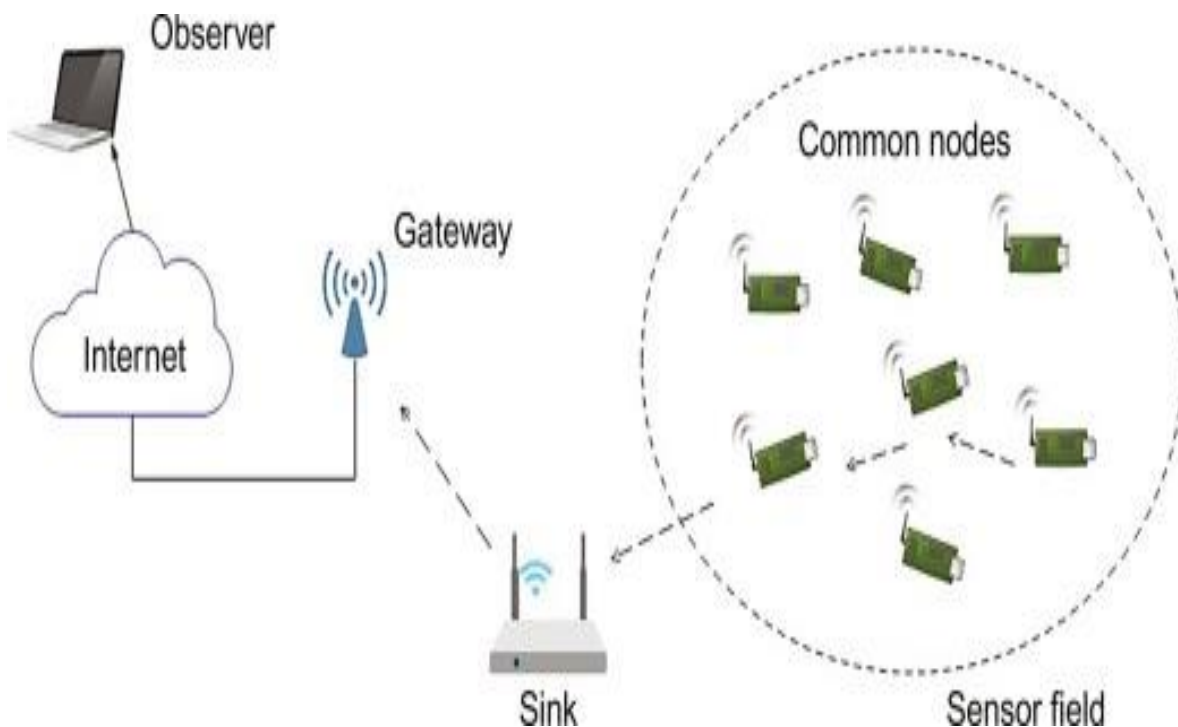


Figure 2: Architecture d'un système basée sur RCSF [2]

3 La caractéristique des réseaux des capteurs

Un ensemble de propriétés importantes pour accomplir les tâches assignées à une application.

Les plus importants :

- **Le type de service** : Le réseau de capteurs sans fils (RCSF) devrait fournir aux utilisateurs des informations significatives sur les objets d'intérêt [1].
- **Scalabilité** : Dans RCSF, un grand nombre de capteurs sont généralement déployés pour assurer une couverture complète de la zone d'intérêt et pour gérer les interruptions, car les capteurs peuvent cesser de fonctionner pour diverses raisons [1].

- **La durée de la vie** : Il s'agit de la durée pendant laquelle le réseau reste actif. Le RCSF doit fonctionner au moins dans le temps requis pour terminer une tâche donnée, cependant, le concept de durée de vie est basé sur le fonctionnement du RCSF et est directement lié aux performances du réseau[1].
- **Maintenance** : Le RCSF doit s'adapter aux changements, s'auto surveiller, adapter son fonctionnement et intégrer d'éventuelles ressources supplémentaires, par exemple des nœuds nouvellement déployés [1].
- **Programmation flexible** : Il s'agit de la capacité des nœuds de capteurs à mettre à jour les alternatives de traitement des données collectées et à apporter des améliorations et des modifications à leurs activités. [1].

4 Architecture physique d'un nœud capteur

Les capteurs sans fil sont des petits dispositifs généralement dotés de quatre composants élémentaires : une unité de capture, une unité de traitement, une unité de communication et une batterie [3] [4].

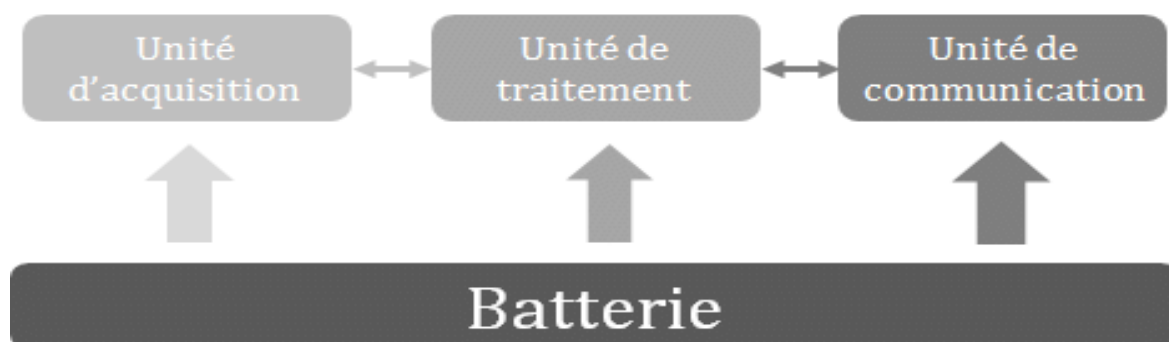


Figure 3:architecture physique d'un nœud capteur [4]

4.1 Unité d'acquisition

L'unité de capture ou d'acquisition, comme son nom l'indique, est chargée de capturer et de transmettre les mesures du phénomène observé à l'unité de traitement. L'ADC (Analog to Digital Converter) gère la conversion du signal analogique de l'unité de capture en un signal numérique interprétable par l'unité de traitement [5].

4.2 Unité de traitement

Les unités de traitement ou transformateurs sont souvent associés à de petites unités de traitement. Stockage. Il récupère les données acquises par l'unité de capture, les analyse, traiter et les transmettre à l'unité de communication [5].

4.3 Unité de communication

Généralement équipé d'un émetteur-récepteur radio, l'appareil est chargé de transmettre et recevoir des données générées par l'unité de traitement, fournissant ainsi aux capteurs la capacité de communiquer avec les autres composants du réseau [5].

4.4 Batterie

Chaque capteur dispose d'une batterie pour alimenter le reste de ses composants. Cependant, en raison de sa taille, il est plutôt limité [5].

5 Problèmes d'énergie dans les Réseaux de capteurs sans fil

Dans un nœud de capteur sans fil Le capteur utilise son énergie pour effectuer trois actions principales: l'acquisition, la communication et le traitement des données :

- **Energie d'acquisition** : Le module de détection absorbe l'énergie de diverses manières, notamment en échantillonnant des signaux physiques et en les convertissant en signaux électriques, en traitement de signal et en conversion analogique-numérique, en raison la grande variété de capteurs, il n'y a pas de valeur standard pour la consommation d'énergie.
- **Energie de traitement** : L'énergie de traitement est classée en deux types : l'énergie de commutation et l'énergie de perte. Au stade du programme, l'énergie de permutation est déterminée par la tension d'alimentation et la puissance globale Permutée (par l'exécution du logiciel), en revanche, l'énergie perdue pendant que la machine de calcul n'effectue aucun calcul est appelée énergie de perte [5].
- **Energie de communication** : qui est normalement un émetteur-récepteur radio, permet au capteur de communiquer avec d'autres capteurs dans un réseau. Il convient de noter que l'énergie de communication représente la majorité de l'énergie utilisée par un nœud de capteur [5].

5.1 Les causes de gaspillage d'énergie

Les réseaux de capteurs sans fil présentent plusieurs défis, dont la plupart entraînent un gaspillage d'énergie, cette section met en évidence les principales causes de gaspillage d'énergie dans la communication des nœuds de capteurs sans fil : [5]

- **Écoute inactive**: Se produit lorsque les nœuds se réveillent et écoutent les trames entrantes même lorsqu'aucune donnée n'est transmise, cela épuise la durée de vie des réseaux de capteurs sans fil [5].

- **Collision** : Elle se produit lorsque deux ou plusieurs stations proches veulent transmettre des paquets en même temps, lorsque cela se produit, tous les paquets impliqués dans la collision doivent être rejetés et retransmis, ce qui entraîne une perte d'énergie [5].
- **Sur-écoute** : Lorsque le canal sans fil diffuse un message, de nombreux nœuds autour de l'expéditeur peuvent entendre la transmission de paquets même s'ils ne sont pas les destinataires prévus de ces transmissions. La sur-écoute d'un trafic inutile peut entraîner un gaspillage d'énergie. [5]
- **Surcharge des paquets de contrôle** : Les paquets de contrôle consomment beaucoup d'énergie lors de l'envoi, de la réception et de l'écoute. Il est donc conseillé d'utiliser un nombre réduit de paquets de contrôle pour la transmission des données afin de réduire la surcharge. [5]

6 Les domaines d'application des réseaux de capteurs

Les réseaux de capteurs sont classiquement déployés comme un emplacement à cause considérer un phénomène spécifique. En fonction d'informations récoltées, ils peuvent participer à plusieurs domaines d'applications tel que : militaires, environnementales, médicales, agricole, domotiques ou plus industrielles [4]. La figure suivant illustre les domaines d'applications possibles.



Figure 4:domaines d'applications de RCFS [4]

7 Domaine agricole (agriculture de précision)

L'agriculture de précision est le principe de gestion des parcelles agricoles visant à améliorer les rendements et les investissements, en s'efforçant de mieux prendre en compte les variations de milieux et de conditions entre les différentes parcelles [6].

Il concerne l'intégration et l'utilisation des technologies de l'information et de divers éléments tels que des capteurs, des systèmes de contrôle, des robots et des véhicules autonomes pour faciliter la gestion des cultures.

Cela peut impliquer l'utilisation de différents types de capteurs qui fournissent régulièrement des informations riches, notamment :

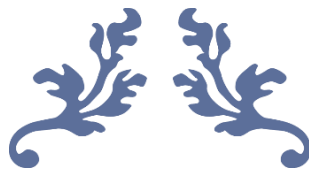
- Capteurs de sol qui collectent des données sur la teneur en azote du sol
- Des capteurs d'irrigation mesurent le niveau d'eau
- Les capteurs d'inondation surveillent également les niveaux d'eau
- Capteurs de surveillance

Les informations fournies par ces capteurs contrôlent automatiquement le système d'arrosage et d'irrigation et alertent automatiquement les utilisateurs lorsque des conditions météorologiques ou d'autres événements se produisent qui peuvent endommager les plantes sensibles.

8 Conclusion

Les réseaux de capteurs sans fil présentent un intérêt considérable et une nouvelle étape dans l'évolution des technologies de l'information et de la communication. Cette nouvelle technologie suscite un intérêt croissant vu la diversité de ces applications : agricole, santé, environnement et industrie.

Dans ce premier chapitre, nous avons présenté les réseaux de capteurs sans fil, leurs architectures de communication, leurs diverses applications et spécialement le domaine agricole ainsi que l'agriculture de précision. Dans le chapitre suivant, nous aborderons sur le problème qui connut un grand essor dans le domaine agricole qui s'appelle les adventices.



Chapitre 02

Généralité sur les mauvaises herbes et la zone
d'étude



1 Introduction

Depuis le début des années 1970, les agriculteurs ont constaté une diminution des rendements des cultures due aux mauvaises herbes et aux ravageurs, ils ont donc été encouragés à utiliser des pesticides et des traitements chimiques pour les éliminer et obtenir les meilleurs résultats malgré leurs dommages environnementaux, ils recherchent donc des solutions intelligentes pour les détecter et les éliminer.

Dans ce chapitre, nous nous intéressons à la définition de cette mauvaise herbe détaillant sa source, ses types, ses avantages et ses inconvénients, et enfin en définir la situation géographique de la région étudiée ainsi que leurs caractéristiques agricole et climatique.

2 C'est quoi une mauvaise herbe

La notion de mauvaises herbes est la notion la plus utilisée dans la profession agricole. Il définit péjorativement des éléments végétaux anachroniques, non semés ou non plantés, et produits par graines ou par reproduction asexuée [7].

Le terme « mauvaise herbe agricole » fait généralement référence aux plantes qui concurrencent sans y être invitées les plantes cultivées. Par conséquent, le terme "mauvaises herbes" est problématique car à moins qu'elles ne soient également toxiques, elles sont elles-mêmes nocives plutôt que nuisibles. C'est pourquoi elles sont assez qualifiées de mauvaises herbes, c'est-à-dire « émergeant de l'extérieur » [8].

Le terme mauvaise herbe a été introduit par les agronomes dès la fin du XVIII^e siècle pour remplacer « mauvaise herbe » [9]. En effet, les espèces végétales adventices peuvent être bénéfiques, neutres ou nuisibles à l'activité humaine, selon le milieu dans lequel elles poussent [10].

3 Source des mauvaises herbes

La plupart des « adventices » sont locales, issues de deux milieux principaux [11]:

- ou un environnement fréquemment perturbé.
- Soit formation végétale en début de succession secondaire.

Actuellement, cependant, avec moins de travail du sol, ces environnements deviennent d'importants fournisseurs de nouvelles espèces de mauvaises herbes [11].

Enfin, on peut considérer un groupe unique d'espèces ex situ, invasives au sens biogéographique du terme, qui n'existent pas dans les formations végétales naturelles indigènes, dont l'introduction remonte à des milliers d'années, mais est plutôt d'origine récente [12].

Ces mauvaises herbes peuvent avoir plusieurs sources et ces espèces peuvent être représentées comme suit [10] :

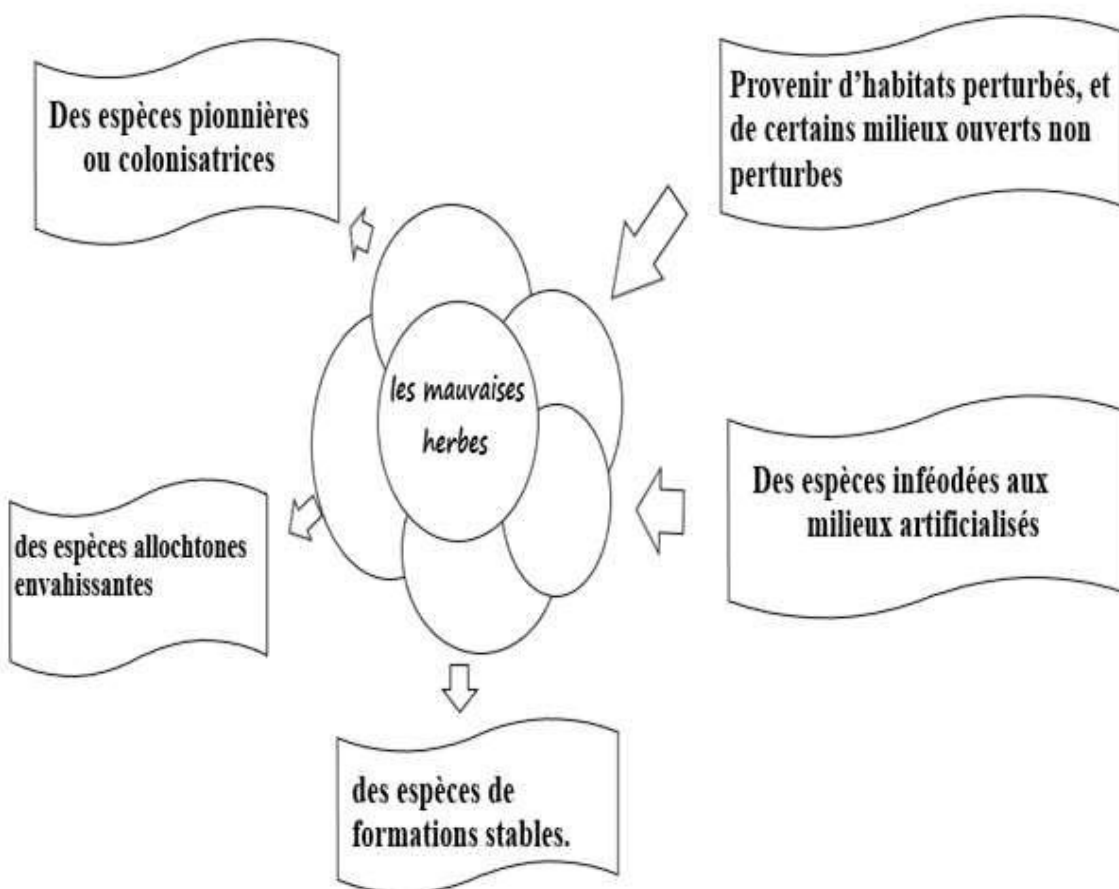


Figure 5:les sources des mauvaises herbes [12]

4 Avantages et inconvénients des mauvaises herbes

4.1 Avantage :

Les herbes compagnes peuvent présenter quelques aspects positifs : [13]

- Formulaires utiles à la protection de la vie pour le jardin (insectes, secours).
- Protège le sol de l'érosion par le vent, la pluie et même du soleil (couverture végétale continue).
- Aère le sol grâce à son système racinaire.
- Présente parfois un réel intérêt esthétique.
- Certaines, comme le chardon ou le lierre, nourrissent et abritent les oiseaux, d'autres, comme le persil sauvage, attirent les insectes.
- Elles éloignent les prédateurs et les maladies.

4.2 Inconvénients [14]:

- Elles puisent l'eau destinée à d'autres plantes.
- Elles affaiblissent les récoltes.
- Attire ou sert d'abri à certains ravageurs des cultures.
- parfois des allergènes végétaux ou des allergies.
- Réduction de la qualité des terres arables.
- Plantes parfois sensibilisées ou sensibilisées.
- La qualité des terres cultivées décline.

5 Les types des mauvaises herbes existant dans Tébessa

En fonction de leur époque préférentielle de germination ou de l'apparition des premières pousses végétatives pour les espèces vivaces.

Les mauvaises herbes ont été réparties dans quatre groupes selon [15] [16]:

Durant toute l'année	Automnale ou hivernale	Printanière	Estival
Géranium sauvage	Scandix pecten-veneris	Parthenium	Digitaria ischaemum
Geranium dissectum	Sherardia arvensis	Papaver rhoeas	Mercurialis annua
Geranium rotundifolium	Aphanes arvensis	Bifora radians	Echinochloa cris-galli
Anthemis arvensis	Ranunculus arvensis	Atriplex patula	Digitaria sanguinalis
Matricaria recutita	Veronica hederifolia	Chenopodium album	Cynodon dactylon
Marguerite commune	Vulpin des champs	Segetum	L'oxalis
Raphanus raphanistrum		Cirsium arvense	
Rumex crispus		Tussilago farfara	
Capselle bourse-à-pasteur		Kickxia spuria	
Agrostis stolonifera		Le liseron des champs	
Herbe du Siam		Le pissenlit	
Poa trivialis		Le chiendent	
Lolium multiflorum		Le chardon des champs	
		Crambe maritima	

Tableau 1: Classement des mauvaises herbes suivant leur période.

6 Situation géographique de la région d'étude

Notre étude a été menée sur le territoire de la province de Tébessa, située au nord-est de l'Algérie "35°24N, 8°6E", d'une superficie de 13,878 km² (13,87800 ha) et une population estimée à 730 306 habitants en 2015. Il est naturellement associé aux vastes prairies du pays [17].

Elle est limitée : au nord par la wilaya de Souk-Ahras ; à l'ouest par Oum El-Bouaghi et les wilayas de Khenchela ; Depuis la promotion de Tébessa au niveau provincial en 1974, la structure territoriale et l'organisation administrative ont changé, avec des réorganisations et corrections successives portant le nombre de communes à 28, chapeautées par 12 daïras [18].

Quant à l'aspect touristique de cet état, il est célèbre pour ses nombreux sites archéologiques romains, notamment dans la ville de Tébessa, tels que la "Porte Caracalla" [18].



Figure 6: Localisation géographique de la wilaya de Tébessa [18]

7 Caractéristiques de la région

7.1 Relief

De par sa situation géographique, la Wilaya de Tébessa s'étend sur différents domaines physiques [17]:

- Au nord, le domaine de l'Atlas présente des structures plissées dont : le mont Tébessa, dont le sommet atteint 1500 mètres d'altitude (Djebel Osmor), 1472 m (DjebelDyr), 1277 m (Djebel Kmakem) et 1358 m (Djebel Onk) [17].

Le plateau est couvert d'un paysage de végétation (Plateau Damon, Saf-Saf - El-Ouesra, Berzguen) [17].

Haute plaine entourée et encadrée par les reliefs décrits précédemment. On note donc les plaines de Tébessa, Morsott, Mchentel et Bhiret-Larneb [17].

- Au sud, la région saharienne a une structure en forme de plaque constituée du plateau saharien, qui prend sa source dans la boucle sud de l'Atlas saharien (Sud du Djebel Onk, Djebel Ma Abiod) [17].

7.2 Climat

D'après les données climatiques recueillies à la station météo de Tébessa (latitude : 35°28 nord longitude : 08°07 est ; altitude : 820 m) [17].

La province de Tébessa est caractérisée par quatre étages bioclimatiques du nord au sud [17]:

- Le semi-humide (400 à 500 mm/an) s'étend rarement, il ne couvre que quelques îlots et se limite aux sommets de certains reliefs (Djebel Serdies et Djebel Bouroumane) [17].
- Le semi-aride (300-400 mm/an) représenté par la couche inférieure froide et froide couvre toute la partie nord de Wailaya [17]
- Des zones subarides (200 à 300 mm/an) couvrent les plateaux steppiques d'Oum Ali, Saf-Saf El Ouesra, Thlidjene et Bir El Ater [17].
- La région saharienne sèche ou douce (-200 mm/an) commence et s'étend au-delà de l'Atlas saharien et couvre les plateaux de Negrin et de Ferkan [17].

7.3 Agricole

En agriculture, la superficie des terres arables est estimée à : 312 047 hectares, dont 12000 hectares sont utilisés pour l'irrigation et la culture : céréales, arbres, etc. [18]

Les terres agricoles du bassin de Wilaya se caractérisent par une répartition anarchique, en fonction de leurs caractéristiques d'affiliation, de la nature culturelle et climatique de la population [19].

Nous distinguons:

- La céréaliculture représentait 5,55% d'El Malabiod SAU et 8,37% d'Houidjbet, soit 6,87% au total [19].
- L'horticulture maraîchère représentait respectivement 6,41% et 7,72% des deux chefs-lieux de la commune (El Malabiod et Houidjbet) [19].
- Dans le même temps, la culture fourragère s'étend à une plus grande surface. 30,48 % et 39,69 %, soit 34,81 % au total [19].

8 Conclusion

Dans ce chapitre, nous avons abordé l'identification des adventices en déterminant leur définition, origine et types présents dans la zone étudiée et leurs avantages, ainsi que étudié une zone géographique de Tébessa et déterminé ses caractéristiques agronomiques et climatiques.

Dans le chapitre suivant, nous verrons la notion d'IA et en détails de l'apprentissage profond (Deep Learning) et exposer la notion des réseaux de neurones artificiels et ses différents types et leurs principe.



Chapitre 03

État de l'art



1 Introduction

La plupart des approches de détection des mauvaises herbes reposent sur des techniques d'apprentissage automatique ou apprentissage profonds.

Dans cette partie, nous essayons de donner une vue générale sur la notion d'IA en citant les outils et les algorithmes mis en œuvre pour l'IA. Premièrement nous aborderons les techniques de l'apprentissage, bien précisément l'apprentissage automatique.

Ensuite citer ses différents types utilisés lors de la conception d'un système doté d'une intelligence artificielle et parler plus en détails de l'apprentissage profond (Deep Learning), ou nous allons exposer la notion des réseaux de neurones artificiels et ses différents types et leurs principe de fonctionnement afin de mieux comprendre la notion de l'apprentissage profond.

Mais avant de commencer ce chapitre plus en détail, nous allons brièvement signifie que des images peuvent être capturées, en fait, il existe plusieurs techniques Au fil des ans, ils ont fait de grands progrès en matière de précision, en particulier dans deviennent plus démocratiques, deviennent plus abordables et peuvent prendre de plus en plus d'images Précis et facile.

2 L'acquisition d'image

Il y a également eu des progrès dans la capture d'images ou de vidéos dans l'agriculture image globale. Si seulement l'équipement spécialisé et qualitativement limitées, les avancées dans la prise de vue ou la capture d'images se sont améliorées et adaptées aux besoins des différents domaines d'application[20].

Il est également indéniable que la démocratisation de cet outil est l'un des plus marquants développements à ce niveau [20].

2.1 Satellite

Par exemple, les champs peuvent être cartographiés à l'aide de satellites ainsi que le géo-référencement pour des informations utiles dans divers domaines Recherche. Ils sont particulièrement utilisés pour les études qui nécessitent une large couverture de terrain Études agricoles ou de diversité végétale telles que (Luoto, Toivonen, & Heikkinen, 2002) [21].



Figure 7: Exemple d'un système d'acquisition d'image satellite [22]

2.2 Les drones

UAV Plus récemment, la démocratisation et la standardisation de l'usage des véhicules Avion de manœuvre (UAV) (Ballesteros, Ortega, Hernandez et Moreno) ou plus communément connu sous le nom de drones, il bénéficie beaucoup de la recherche en agriculture de précision, Ces véhicules peuvent capturer des images très précises [20].

Drone autorisé y compris des réductions de coûts significatives par rapport à l'utilisation de satellites, et Meilleure résolution des images centimétriques et même millimétriques (UAV) Par rapport aux mètres de l'imagerie satellite [20].

Selon le contexte de l'étude et les exigences du contexte, plusieurs modalités d'imagerie peuvent être ajustées en fonction de la caméra utilisée et des paramètres (image couleur RVB, image niveaux de gris, images infrarouges et thermiques...etc.)[20].

2.3 Caméra

Tirer a fourni une partie de la puissance de la vision artificielle pour l'agriculture reposant sur la sélection de la bonne caméra. Tout d'abord, il est important d'identifier le problème à résoudre car cela a déterminé la région de longueur d'onde, la résolution requise, la vitesse et l'interface de la caméra [20].

Les conditions environnementales pour l'imagerie seront également importantes. L'imagerie dans la plage visible avec des conditions d'éclairage variables nécessite une caméra avec une plage dynamique élevée pour capturer des images sans sursaturation ni perte de détails dans le bruit de lecture [20].

2.4 L'utilisation des téléphones intelligents

L'objet technologique le plus vendu aujourd'hui est sans conteste le téléphone intelligent. Ces appareils embarquent des caméras avec de plus en plus de capteurs puissants. Il est donc naturel de vouloir exploiter ce pouvoir. Alors appelle Pour certaines tâches de capture, les téléphones portables sont le remplacement parfait pour d'autres appareils Les images et les commandes utilisent leurs autres avantages, tels que le GPS intégré, leur Connectique (GSM, WIFI, Bluetooth...etc.) [20].

3 Notion d'Intelligence artificielle

L'intelligence artificielle (IA) renvoie à "une application capable de traiter des tâches qui sont, pour l'heure, réalisées de manière plus satisfaisante par des êtres humains dans la mesure où elles impliquent des processus mentaux de haut niveau comme l'apprentissage perceptuel, l'organisation de la mémoire et l'esprit critique"[23].

C'est ainsi que le scientifique américain Marvin Lee Minsky, considéré comme le père de l'IA, définit ce concept. C'est en 1956 à l'occasion d'une rencontre de scientifiques à Dartmouth (au sud de Boston) organisée en vue d'envisager la création de machines pensantes qu'il parvient à convaincre son auditoire d'accepter le terme [23].

Les scientifiques et afin de créer une intelligence artificielle, utilisent un ensemble de techniques différentes regroupées sous une même discipline qu'on appelle l'apprentissage machine (ou machine Learning, en anglais).

Donc dire intelligence artificielle revient automatiquement à dire "Machine Learning", ou l'apprentissage automatique [23].

4 L'apprentissage automatique (Machine Learning)

L'apprentissage automatique est considéré comme un sous-domaine de l'intelligence artificielle. Son objectif majeur est de comprendre la structure des données et de les faire intégrer dans des modèles simples que tout le monde peut comprendre et utiliser [24].

Bien que l'apprentissage automatique soit un domaine de l'informatique, il diffère de ces approches traditionnelles. [24]

De nos jours, l'apprentissage automatique est omniprésent dans toutes nos interactions avec les banques, nos achats en ligne et notre utilisation des médias sociaux. [24]

Il existe plusieurs méthodes d'apprentissage automatique, mais les plus largement utilisées sont l'apprentissage supervisé ou les algorithmes sont basés sur des données d'entrées et de sorties bien spécifiées par l'homme et l'apprentissage non supervisé qui ne fournit pas à l'algorithme des données étiquetées afin de lui pouvoir permettre de définir une structure et de découvrir une logique dans les données entrées. Auxquelles on ajoute l'apprentissage par renforcement et l'apprentissage profond. Explorons donc ces méthodes plus en détail. [24]

4.1 L'apprentissage supervisé

L'apprentissage supervisé est une variété de machine Learning qui utilise un ensemble de données d'apprentissage étiquetées afin de créer des modèles d'intelligence artificielle. Le but de cette méthode c'est le fait que l'algorithme soit capable «d'apprendre» en comparant sa sortie réelle avec les sorties «enseignées» pour trouver des erreurs et modifier le modèle en conséquence. Donc l'apprentissage supervisé utilise des modèles pour déterminer les valeurs d'étiquettes pour un ensemble de données non étiquetées. [24]

Dans le cadre de l'apprentissage supervisé, l'algorithme est entraîné par un ensemble de données qui est déjà étiqueté et qui a une sortie prédéterminée. [24]

4.2 L'apprentissage non supervisé

Dans le cas de l'apprentissage non supervisé, on utilise un ensemble de données d'entrées non étiquetées, afin de laisser l'algorithme d'apprentissage trouver tout seul les points communs parmi cet ensemble de données. Les méthodes d'apprentissage automatique qui facilitent l'apprentissage non supervisé sont particulièrement utiles, vu que les données non étiquetées étant plus considérables que celles étiquetées [24].

On peut considérer que l'objectif initial de l'apprentissage non supervisé est aussi simple que de détecter les modèles cachés dans un ensemble de données, mais il peut aussi avoir un objectif d'apprentissage des caractéristiques, ce qui va rendre la machine intelligente capable de découvrir automatiquement les représentations nécessaires pour classer des données brutes[24].

4.3 L'apprentissage par renforcement

L'apprentissage par renforcement fait référence à une classe de problèmes d'apprentissage automatique, qui consiste à apprendre à partir d'expériences successives, ce qu'il convient de faire de façon à trouver les meilleures solutions. [24]

Autrement dit, les machines intelligentes essaient plusieurs situations afin de pouvoir déterminer les actions les plus avantageuses, et ne se contentent pas de recevoir des instructions sur les actions à appliquer, ce qui distingue cette méthode des autres techniques d'apprentissage. [24]

4.4 L'apprentissage profond

L'apprentissage profond (dl), est l'une des principales technologies de Machine Learning et d'intelligence artificielle. Découvrez en quoi consistent cette technologie, son fonctionnement, et ses différents secteurs d'application dans les sous titres suivants ou tout sera détaillé [24].

5 Apprentissage Profond (Deep Learning)

5.1 Définition et origine

Le Deep Learning ou apprentissage profond est un type d'intelligence artificielle, dérivé du machine Learning qui a été développé dans le but de créer des algorithmes capables d'apprendre et de s'améliorer de manière autonome, contrairement à la programmation où la machine se contente d'exécuter à la lettre des règles prédéterminées [24].

Profonde de la formation est pas seulement une question de technologie, mais aussi d'une vision globale de l'homme. L'un des premiers objectifs d'un apprentissage en profondeur est la lecture des capacités de l'homme et même la création d'une version améliorée [24].

Profondément apprentissage est considéré comme un mode d'apprentissage automatique, géré par l'artificiel d'un réseau neuronal. Le réseau est composé de plusieurs couches avec des fonctions précises [24].

Par conséquent, la plus profonde de la formation vise à la modélisation du cerveau humain, l'activation de ces neurones des couches, qui interagissent et fournissent progressive de la formation sur la base de grandes quantités de données. Par conséquent, comme les neurones

dans le cerveau, les neurones peuvent communiquer, la promotion de l'apprentissage et la maîtrise des différents éléments d'information. Découvrez en quoi consistent cette technologie, son fonctionnement, et ses différents secteurs d'application dans le titre suivant ou tout sera détaillé [24].

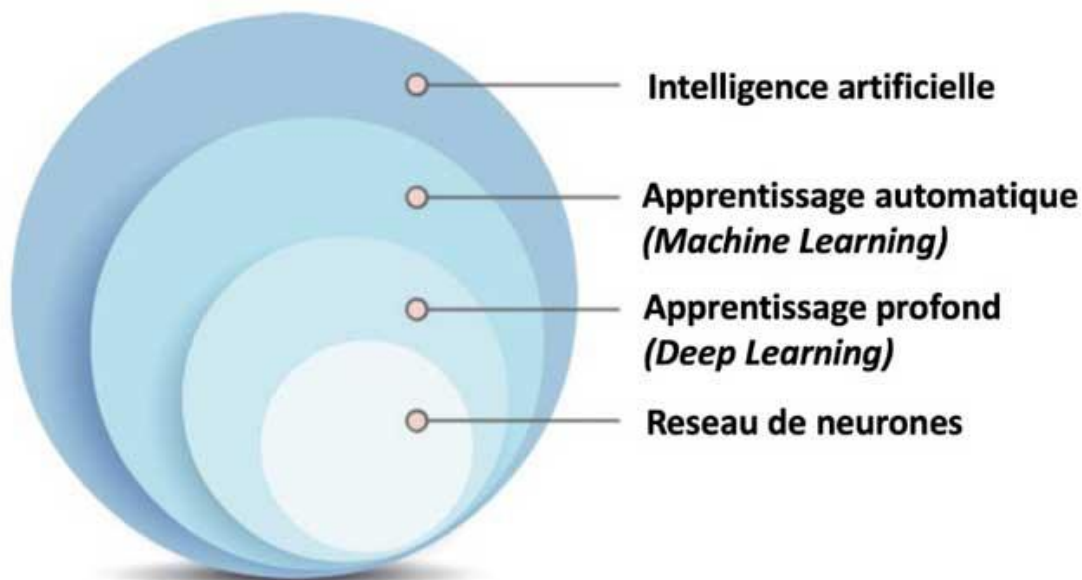


Figure 8: La relation entre l'IA, apprentissage automatique et profond ainsi que réseau de neurone [61]

5.2 Les réseaux de neurones

”Les réseaux de neurones artificiels sont des réseaux fortement connectés de processeurs élémentaires fonctionnant en parallèle. Chaque processeur élémentaire calcule une sortie unique sur la base des informations qu’il reçoit. Toute structure hiérarchique de réseaux est évidemment un réseau.” [25] [32]

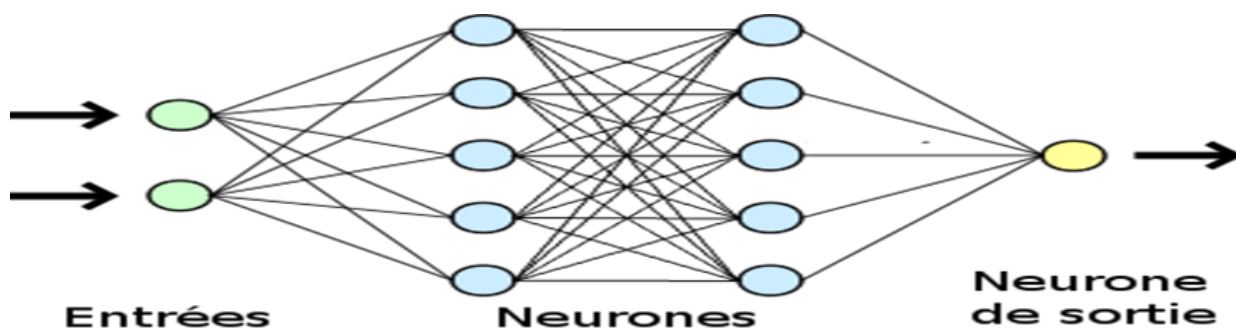


Figure 9: Exemple d'un réseau de neurones artificiel

5.3 Quelque type d'algorithmes utilisés en Deep Learning

Les architectures les plus utilisées en réseaux de neurones profonds sont : [27]

5.3.1 Les réseaux de neurones profonds (DNN)

Les réseaux de neurones profonds (Deep Neural Networks : DNN) sont des réseaux à flux direct dans lesquels les données circulent de la couche d'entrée à la couche de sortie sans revenir en arrière et les liens entre les couches sont à sens unique et ils ne touchent jamais un nœud de nouveau [27].

5.3.2 neurones récurrents (RNN)

Un réseau neuronal récurrents (Recurrent Neural Network : RNN) est une classe de réseaux neuronaux artificiels dans lesquels les connexions entre les nœuds forment un graphique vectoriel le long d'une séquence qui leur permet de montrer un comportement dynamique temporel d'une séquence temporelle. Les RNN peuvent utiliser leur état interne (mémoire) pour gérer la séquence d'entrée. Cela le rend applicable à des tâches telles que la reconnaissance de l'écriture manuscrite ou la reconnaissance vocale. [27]

5.3.3 Les réseaux récurrents à mémoire court et long terme (LSTM)

Les réseaux récurrent a mémoire court et long terme (Long short-term memory : LSTM) sont un type particulier de RNN, capable d'apprendre les dépendances à long terme qui permettent à RNN de se souvenir de choses qui se sont produites dans le passé et de trouver des modèles à travers le temps pour donner un sens à ses prochaines suppositions. [27]

Les LSTM ont battu des records pour l'amélioration de la traduction automatique, de la modélisation linguistique et du traitement linguistique multilingue. [27]

5.3.4 Les réseaux antagonistes génératifs (GAN)

C'est le profond algorithme d'apprentissage derrière le générative de l'IA. Entre autres choses, Gan peut créer de nouvelles données de cas qui sont similaires aux données qu'ils ont été formés. Les réseaux de génératives antagonistes se composent d'un discriminateur de générateur. Ils sont utilisés à des fins diverses, telles que la création ou l'amélioration des images, la génération de texte, etc. [27]

5.3.5 Les réseaux de neurones Convolutif (CNN)

Les réseaux de neurones Convolutif (CNN ou ConvNets) sont une classe de réseaux de neurones profonds couramment appliqués à l'analyse d'images visuelles. Leurs autres applications incluent la compréhension vidéo, la reconnaissance vocale et la compréhension du traitement du langage naturel. [27]

6 Le Réseau Neuronal Convolutif (CNN)

6.1 Définition

Le Réseau Neuronal Convolutif (CNN) est comparable à un réseau de neurones standard. Il est aussi composé de neurones avec différentes opérations simples [28].

Cependant, les CNN présument que les données entrantes sont en forme de grille ou matrice. Ce type de données se présente dans plusieurs domaines : longueur d'onde audio [29], le traitement automatique du langage naturel [30] ou plus communément des images [17].

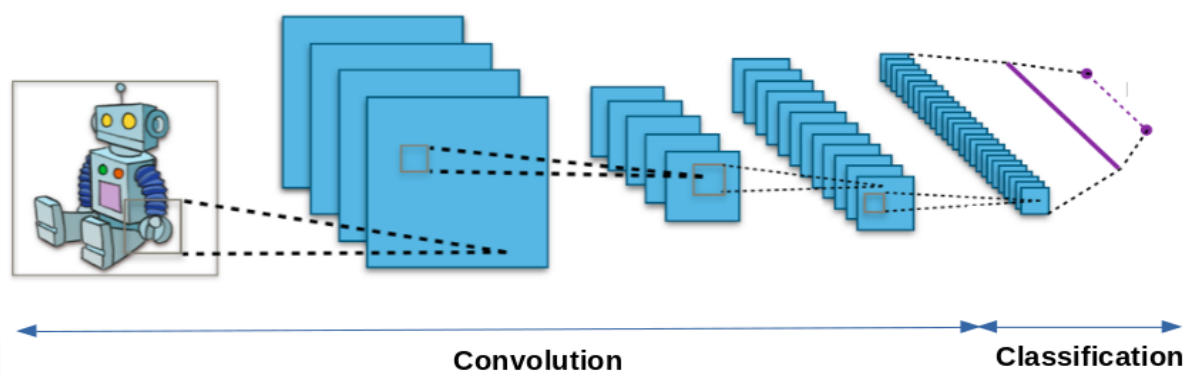


Figure 10: Architecture d'un réseau de neurones Convolutif [32].

6.2 Les couches de CNN

Une architecture d'un réseau Convolutif se compose de 4 types de couches différentes :

6.2.1 Couche Convulsive

Elle consiste à appliquer un filtre de convolution à l'image pour détecter des caractéristiques de l'image. Une image passe à travers une succession de filtres, ou noyaux de convolution, créant de nouvelles images appelées cartes de convolutions. Certains filtres intermédiaires réduisent la résolution de l'image par une opération de maximum local. Au final, les cartes de convolutions sont mises à plat et concaténées en un vecteur de caractéristiques [33].

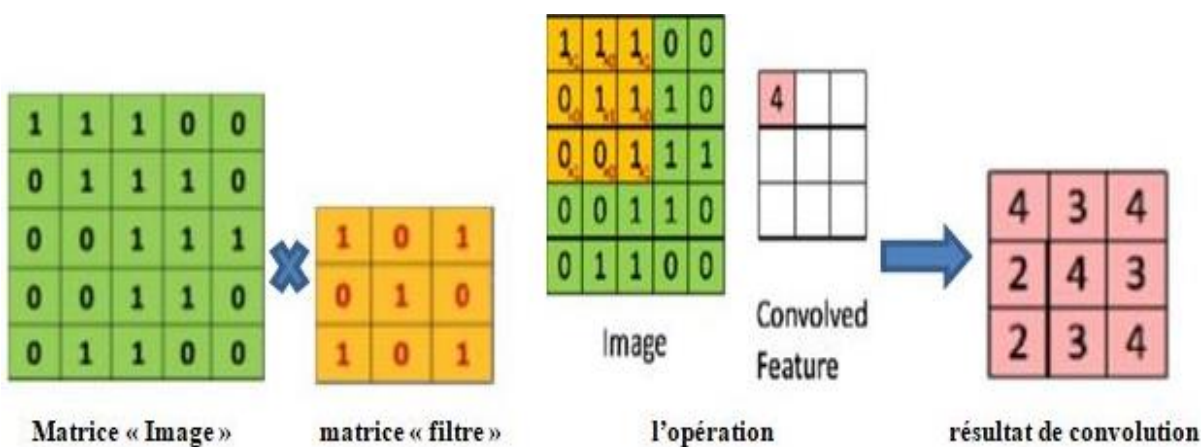


Figure 11:Schéma du parcours du résultat de convolution [33]

6.2.2 Couche de correction ReLU

L'on fait passer les cartes de convolution à travers une couche d'activation non linéaire telle que Rectified Linear Unit (ReLU), qui consiste à remplacer les nombres négatifs des images filtrées par des zéros [33].

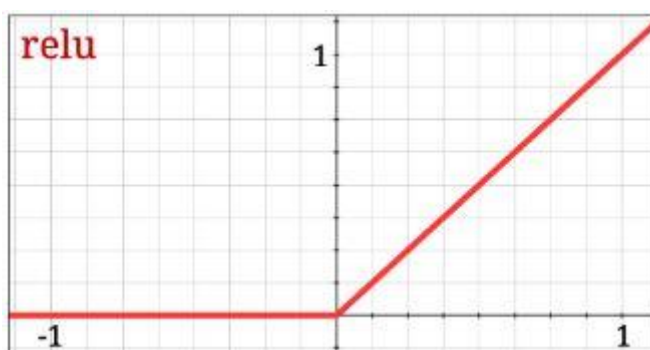


Figure 12: Rectified Linear Unit (ReLU) [33]

6.2.3 Couche de pooling

Consiste à réduire progressivement la taille de l'image en ne gardant que les informations les plus importantes, par exemple pour chaque groupe de 4 pixels, le pixel ayant la valeur maximale (Max Pooling, le plus populaire) ou la moyenne des pixels (AVG pooling) [33].

Avec la couche pooling, la quantité de paramètres et de calcul dans le réseau sont réduits, et cela va permettre de contrôler le sur-apprentissage [33].

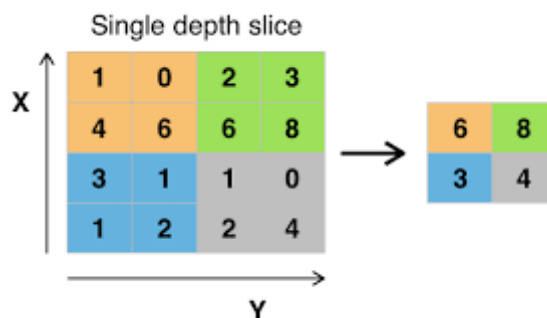


Figure 13: la couche max pooling [33]

Après avoir extrait les caractéristiques des entrées, on attache à la fin du réseau un perceptron ou bien un MLP (Multi Layer perceptron) communément appelé Fully connected [33].

6.2.4 Couche entièrement connectée (fully-connected)

Ce type de couche reçoit un vecteur d'entrée et produit un nouveau vecteur de sortie. Pour cela, il applique une combinaison linéaire et éventuellement une fonction d'activation aux valeurs reçues en entrée [33].

7 Réseaux de neurones de convolution pour la détection d'objets

7.1 R-CNN

Le R-CNN est une architecture de détection d'objet. Le R-CNN commence par extraire des régions intéressantes de l'image, puis il utilise ces régions comme données d'entrée pour un CNN [35].

Cette séparation en régions permet de détecter plusieurs objets de plusieurs classes différentes dans une même image. Cette solution proposée par Girshick et al. 2013 a permis d'améliorer la précision des modèles de détection [34].

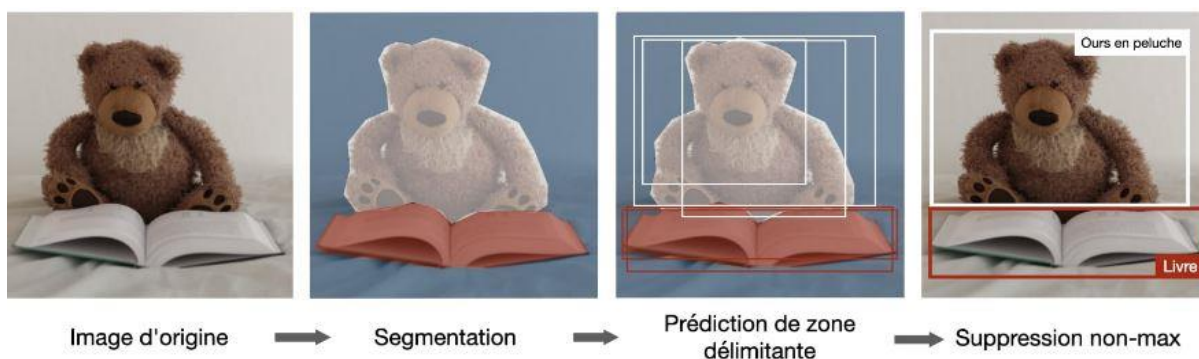


Figure 14: exemple d' Architecture R-CNN [36]

7.2 Yolo

YOLO est l'abréviation du terme "You Only Look Once". Il s'agit d'un algorithme qui détecte et reconnaît divers objets dans une image (en temps réel). La détection d'objets dans YOLO se fait comme un problème de régression et fournit les probabilités de classe des images détectées [35] [37].

L'algorithme YOLO utilise des réseaux de neurones convolutionnels (CNN) pour détecter des objets en temps réel. Comme son nom l'indique, l'algorithme ne nécessite qu'une seule propagation vers l'avant à travers un réseau de neurones pour détecter des objets [37].

7.3 Faster R-CNN

Faster R-CNN est une amélioration du R-CNN dans sa précision et sa rapidité à l'entraînement (Ren, He, Girshick & Sun, 2015) [38]

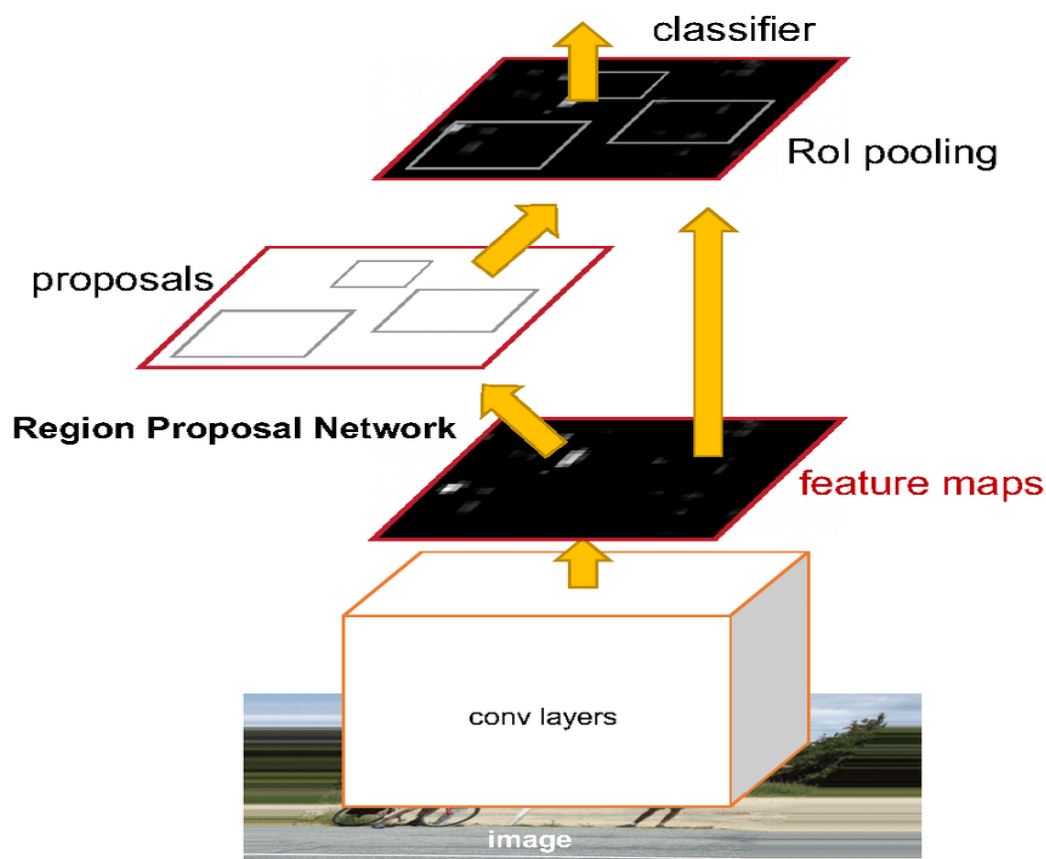


Figure 15: Architecture Faster R-CNN [38].

7.4 Mask R-CNN

Le Mask R-CNN est l'amélioration suivante après le Faster R-CNN. Le Mask R-CNN change la sortie du modèle final. En effet, l'architecture Faster R-CNN permet de localiser des objets distincts avec une boîte englobante.

L'architecture Mask R-CNN permet, quant à elle, de segmenter chaque instance d'un objet avec un masque sémantique. Cette amélioration permet donc au modèle de faire de la segmentation d'instance. [38]

L'architecture Mask R-CNN est semblable à Faster R-CNN mais rajoute des couches en parallèle de la classification. Cette approche est différente d'autres architectures de segmentation d'instance qui classifiaient les résultats par rapport au masque génère (39).

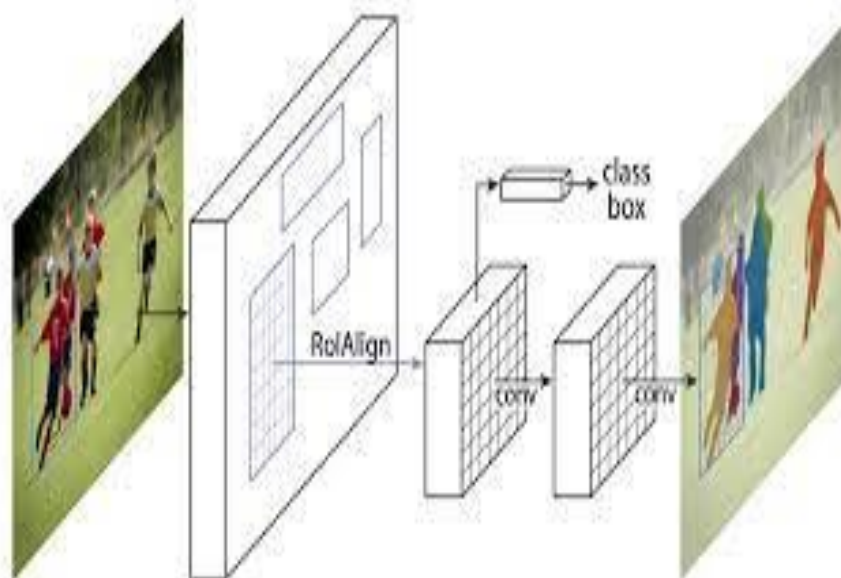


Figure 16:l'architecture de modèle MaskRCNN [39]

8 Les méthodes utilisées pour détecter les mauvaises herbes

À ce jour de plus en plus de recherches sont en cours dans le seul but d'améliorer et adopter de meilleures techniques pour faciliter la détection des mauvaises herbes et contrôle.

L'approche majeure et commune appliquée dans les recherches est la méthode de détection par traitement d'image et techniques de filtrage. Dans ce tableau on décrit une collection de divers travaux de recherche rapportés par des chercheurs sur la détection des adventices [40].

Contributeurs	méthodes	descriptions
Latta et al [41]	clustering K-means	Ont proposé cette méthode pour détection des mauvaises herbes en utilisant la technique inter-rangée et inter-plante pour détecter et contrôler les mauvaises herbes sur un champ de maïs avec des résultats variables à chaque fois la méthode était express.[41]
Michel et al [42]	combinant la technique de traitement d'image et l'étiquetage des composants	Dans leurs recherches ont détecté des mauvaises herbes par combinant la technique de traitement d'image et l'étiquetage des composants méthode avec rapport d'aspect pour comparer les hauteurs et la largeur des images pour la détection et le contrôle des mauvaises herbes [42].
Shraddha et al [43]	développée système se compose d'un émetteur récepteur RF qui transmet sans fil signaux basés sur l'entrée reçue à l'aide de MATLAB	Une méthode de détection et de contrôle des mauvaises herbes a été développée et mis en œuvre. L'intelligence développée système se compose d'un émetteur-récepteur RF qui transmet sans fil signaux basés sur l'entrée reçue à l'aide de MATLAB pour filtrer et séparer la couleur des mauvaises herbes à un pulvérisateur robotisé pour une application efficace [43].
Radhika et Roopa [44]	présenté une vision artificielle système de détection de mauvaises herbes	Présenté une vision artificielle système de détection de mauvaises herbes dans les cultures maraîchères par filtrage d'images pour extraire les caractéristiques de couleur et de surface en accordant une attention particulière à le niveau d'éclairage de la zone capturée pour faciliter une analyse efficace [44].
Sandep et al [45]	combinaient les procédures de segmentation avec Réseau de neurones Convolutif (CNN) et	pour séparer les mauvaises herbes des cultures dans une plantation de carottes. la collecte de données RVB sur les mauvaises

	traitement d'images technique	herbes a été conservée dans une base de données et mis en œuvre avec le langage de programmation Python [45]. La méthode aboutie à une précision de 90 % dans la classification adoptée par CNN et détection des mauvaises herbes [45].
Adel et Abdoulabbas [46]	Techniques d'apprentissage automatique - Support Vector Machine et Réseaux de neurones artificiels	ont intégré plusieurs caractéristiques de forme avec des techniques d'apprentissage automatique - Support Vector Machine et Réseaux de neurones artificiels pour aider à la détection précise des mauvaises herbes sur un champ de betteraves sucrières [46]. Les résultats obtenus reflètent fidèlement l'ANN Adopté correctement classé 92,5% de la mauvaise herbe à une précision niveau de 92,9 %.
Grinblat et al [47].	Deep Convolutional Réseau de neurones	ont identifié des espèces de légumineuses à l'aide de Deep Convolutional Réseau de neurones pour classer les nervures des feuilles sur la morphologie Motifs [47].
Xu et al [48]	un système optimisant Absolute Feature Corner Point (AFCP)	L'algorithme a fusionné les coins absolus pour identifier les positions des cultures et des mauvaises herbes. Paramètres clés de ce modèle sont la pression des mauvaises herbes et le taux de grappes. Validation effectuée sur la base les images en temps réel capturées sur les terres agricoles ont atteint une précision niveau de 90%, révélant ainsi que l'algorithme AFCP est applicable pour la détection et la gestion des mauvaises herbes en temps réel [48].
Femi Temitope Johnson et Joel	un aperçu de l'application de l'intelligence artificielle	donne un aperçu de l'application de l'intelligence artificielle (IA) comme le meilleur approche techno-efficace pour la détection des mauvaises herbes Résultats d'exécution réalisés

Akerele [49]		avec MATLAB, Il a donné une précision de détection de quatre-vingt-quinze pour cent (90%).De même, la classification Supporting Vector Machine (SVM) Approuvé pour la détection des mauvaises herbes a également donné (90%) [40].
--------------	--	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Tableau 2:les travaux de recherche rapportés par des chercheurs sur la détection des adventices

8.1 Synthèses

Les mauvaises herbes ont eu un effet profond sur l'agriculture dans le monde entier et le nombre de perte de récolte agricole continue de croître dans le monde. Bien que la technologie ait pénétré notre vie quotidienne avec un grand succès, en particulier en ML et DL, l'IA a également contribué à détecter ces mauvaises herbes pour éliminer les soit traditionnellement ou par machine et Sensibiliser les agriculteurs à se protéger des méfaits des pesticides et les produits chimique et réduisent leur utilisation.

DL n'est qu'un moyen efficace de fournir des solutions prometteuses axées sur les données pour aider l'agriculteur à détecter les mauvaises herbes. Ils proposent plusieurs méthodes de détection des mauvaises herbes dans ces méthodes de regroupement K-means, en utilisant des techniques inter-rangées et inter plantes pour détecter et contrôler les mauvaises herbes.

D'autre part, k-means a des difficultés à regrouper les données lorsque la taille et la densité des grappes sont différentes, ou à incorporer des techniques de traitement d'images pour comparer la hauteur et la largeur des images pour la détection et le contrôle des mauvaises herbes. L'inconvénient est qu'il ne donne pas de résultats clairs.

Nous avons étudié les techniques les plus appropriées pour répondre aux besoins des agriculteurs et Capable de s'adapter rapidement à de nouveaux problèmes et domaines. L'apprentissage en profondeur nous donne Meilleure précision de reconnaissance et fonction d'apprentissage à partir de chaque image envoyée par les capteurs. Nous avons étudié d'autres technologies telles que k-means, SVM avec Fonctionnalités extraites manuellement, etc. Mais ils ne répondaient pas à nos besoins... et aujourd'hui, Nôtre vision est de continuer à développer l'application afin que l'analyse puisse être fournie Pour un champ observé de différents côtés en temps réel.

9 Conclusion

Dans ce chapitre, on vu la notion d'IA et en détails de l'apprentissage profond (Deep Learning) et exposer la notion des réseaux de neurones artificiels et ses différents types et leurs principe car 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. Dans le chapitre suivant, nous verrons la présentation de l'architecture globale du système.



Chapitre 04

Contribution



1 Introduction

Nous avons introduit dans le chapitre précédent les différentes techniques utilisées dans le traitement d'images et plus particulièrement la détection dans le domaine de l'agriculture de précision.

Les dernières années ont connu un regain énorme de l'utilisation des techniques de détection 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.

Le présent chapitre sera dédié à la présentation de l'architecture de notre système ainsi que les mauvaises herbes étudié pour cette contribution.

2 Les mauvaises herbes

Il y a sans aucun doute un grand nombre de mauvaises herbes dans la nature, et certaines peuvent, ils sont très utiles dans un contexte, mais sont considérés comme nuisibles dans un autre, dans le nôtre Les recherches sur lesquelles nous nous sommes naturellement concentrés sont les mauvaises herbes trouvées à Tébessa, qui affectent grandement les cultures et sont donc considérées comme des mauvaises herbes.

Dans nos recherches, nous avons mis en lumière les espèces les plus abondantes et les plus nuisibles tels que les chardons des champs Le pissenlit, Le chiendent avec leurs types etc....



Figure 17:Les mauvaises herbes étudiées (extrait de la base de données)

3 L'acquisition des images

Aujourd'hui, la chose technologique la plus vendue et la plus utilisée est sans aucun doute le smartphone. Ces appareils incluent des caméras avec des capteurs de plus en plus puissants, ainsi le désir de profiter de ce pouvoir est naturel.

C'est pourquoi nous avons obtenu notre base de données d'images de mauvaises herbes en les capturant avec notre smartphone pour obtenir des images de haute qualité et faciliter la détection de ces mauvaises herbes.

4 Conception globale

Dans cette section, nous allons essayer de donner une vue générale de notre système qui consistera à insérer une nouvelle image envoyé par les capteurs sans fils installer et à détecter les mauvaises herbes existant dans ces images.

Dans notre conception d'application on met plusieurs nœuds capteurs sont d'employés dans un certain champ agricole ou massif pour prendre des photos et faire remonter les informations collectées à une station de base, nommée le nœud puits (une porte d'entrée vers le monde extérieur qui fait l'interface entre le réseau de capteurs et l'utilisateur des données).

Dans le cas le plus simple, les capteurs seront dans le voisinage direct du puits (un réseau de type 'Etoile a un saut). Cependant, dans le cas d'un réseau à grande échelle, les capteurs ne sont pas tous dans le voisinage du puits et les messages seront acheminés du nœud source vers le puits en transitant par plusieurs nœuds, selon un mode de communication multi-sauts comme l'illustre dans la figure

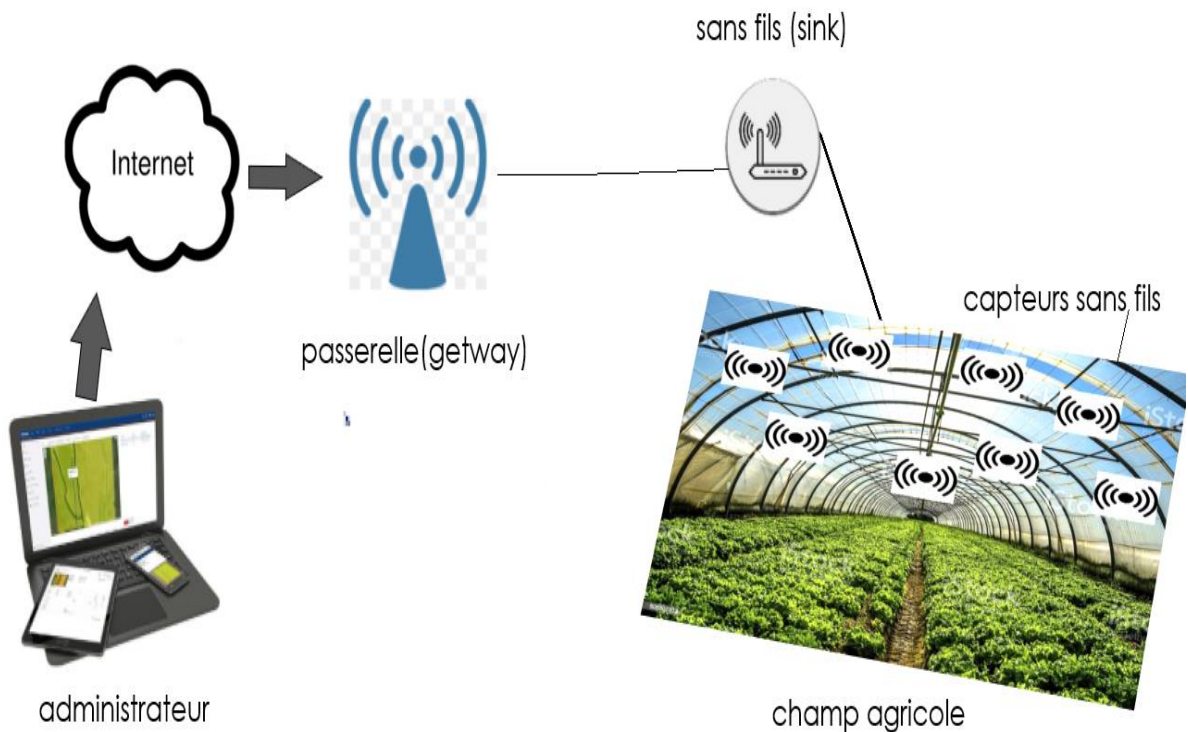


Figure 18: Conception de notre système basé sur RSCF

Lorsque l'administration recevra les images envoyées par les capteurs, elle les traitera à l'aide d'un modèle basé sur apprentissage profonds spécialement les réseaux neurones Convolutif pour détecter les mauvaises herbes existant dans ces images pour éliminer le, et obtenir des bon récoltes sur apprentissage profonds spécialement les réseaux neurones Convolutif pour détecter les mauvaises herbes existant dans ces images pour éliminer le et obtenir des bon récoltes.

Notre approche de detection a passé par plusieurs étape en commençant par le traitement de donnée nous nous traiterons notre données "images" qui seront divisées en deux parties 80% données d'entraînement et 20% données de validation et passeront par la phase de prétraitement, comme illustré à la figure suivante. Ensuite, nous créons un modèle d'apprentissage qui prend les entrées de données traitées à l'étape passée et s'entraîne sur une partie des données (données d'entraînement).

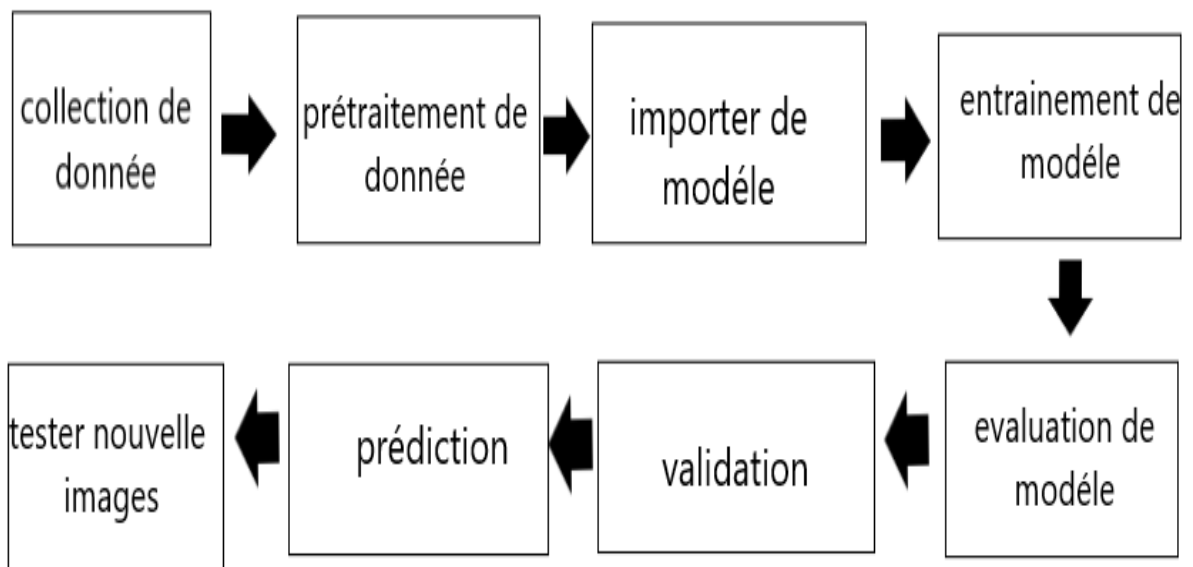


Figure 19: La conception global du notre projet

4.1 collection de donnée

Après avoir abordé dans le premier chapitre le concept des mauvaises herbes présentes à willaya de Tébessa, leurs types et leur origine, nous avons délibérément pris des photos de haute qualité de toutes tailles et directions sans brouillard pour les espèces les plus abondantes et les plus connues en termes de forme et de présence.

4.2 prétraitement de donnée

Après avoir sélectionné l'ensemble de données, une étape très importante dans toute analyse d'image est le traitement d'image. Les images doivent être traitées et modifiées de manière à être pleinement compatibles pour obtenir une plus grande efficacité du modèle conçu.

Pour alimenter les images d'entrée dans le formulaire, les images sont normalisées pour une meilleure lisibilité. Les images ont été reconfigurées selon les exigences de la forme construite.

4.2.1 Division de donnée

L'ensemble de données contient 200 images sur 858 x 644 enregistrées au format JPG, dans lequel 183 images ont été utilisées pour la formation et 19 images pour la validation. On vu Des exemples de ces photos sont présentés dans la figure suivante :



Figure 20: Echantillon de donnée

4.2.2 Limiter les mauvaises herbes

Nous allons former deux modèle de detection des mauvaise herbes, donc nous créons des annotations de masque au niveau des pixels pour définir les limites des objets dans l'ensemble de données.

Parmi les différents outils disponibles, nous choisissons deux outils intuitif et bien fait : VGG Image Annotator (VIA) pour le modèle Mask R-CNN et le roboflow pour le yolov5. VIA ne nécessite aucune installation, il vous suffit de télécharger le package et d'ouvrir le fichier via.html avec un navigateur moderne ainsi que le deuxième outil c'est un site existe et facile d'utilise le.



Figure 21: limitation des mauvaises herbes

4.3 Exportation des données

Lorsque l'annotation est terminée pour toutes les images, enregistrez les annotations en tant que fichier sous format JSON ou du format yolov5 pytorch pour le modèle yolov5. On travaille séparément pour les ensembles de données d'apprentissage et de test.

4.4 Importation de modèle

Pour vu un résultat bien défini, nous nous fait deux modèles de detection et on comparer entre leurs résultats et trouver le meilleure à partir de leurs prédiction et leurs résultat. Dans la section suivant on détaillants chaque architecture. On commençant par le modèle Mask RCNN 2017 :

4.4.1 Mask R-CNN 2017

MaskRCNN ajoute une branche de prédiction de masque à la dernière étape de FasterRCNN. Dans la première étape est constituée de deux réseaux, backbone (ResNet 101) et RPN. Ces réseaux fonctionnent une fois par image pour donner un ensemble de propositions de régions. Les propositions de région sont des régions de la carte d'entités qui contiennent l'objet.

Dans la deuxième étape, le réseau prédit box et la classe d'objets pour chacune des régions proposées obtenues à l'étape 1. Chaque région proposée peut être de taille différente alors

que les couches entièrement connectées dans les réseaux nécessitent toujours un vecteur de taille fixe pour faire des prédictions. La taille de ces régions proposées est fixée en utilisant l'un ou l'autre des pools RoI.

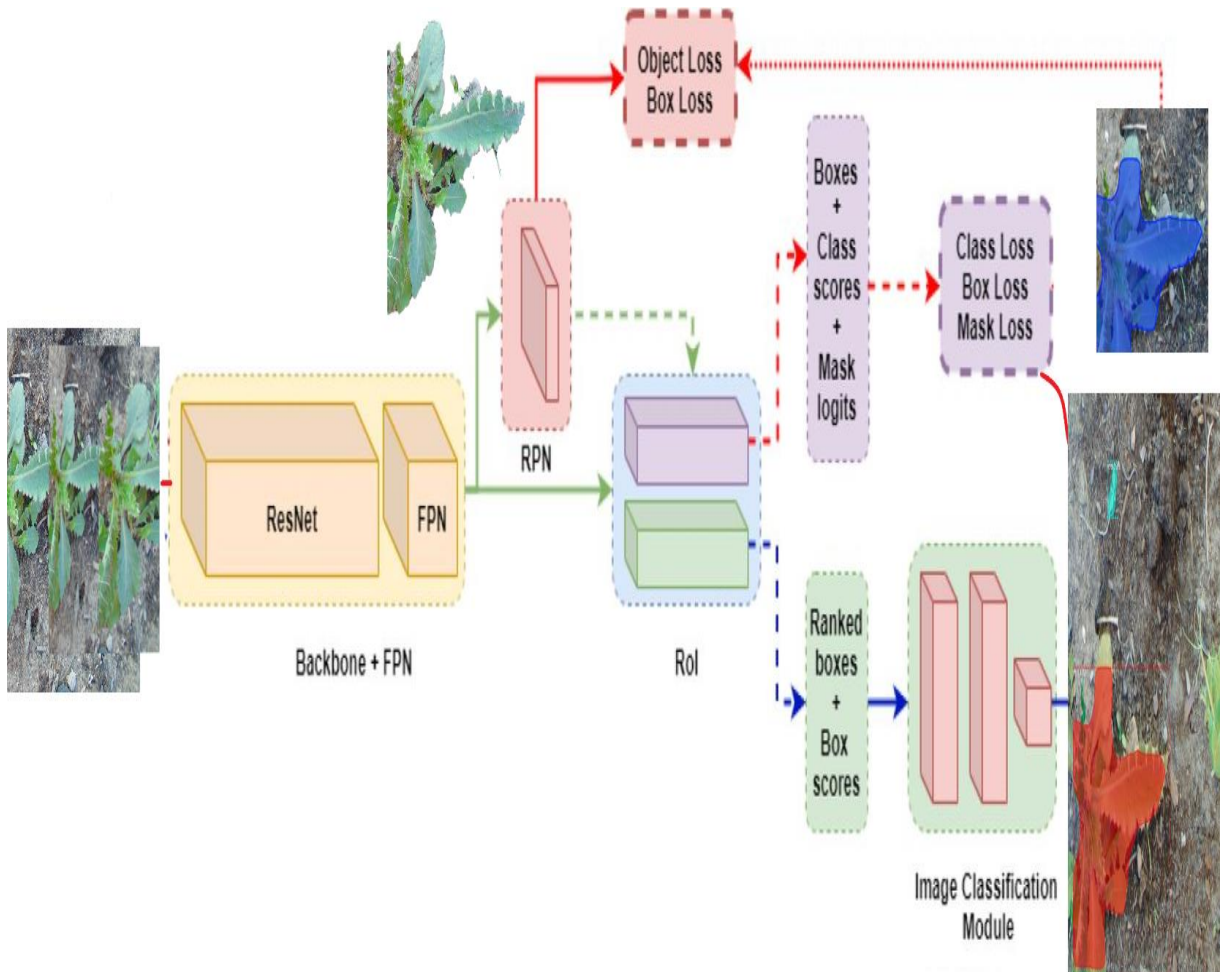


Figure 22:l'architecture de detection les mauvaises herbes basée sur le Mask R-CNN
Cependant, l'architecture FPN utilise des caractéristiques de plusieurs couches de convolution et nous pouvons en choisir une ou une combinaison. Cette hiérarchie de FPN donne une meilleure prédiction.

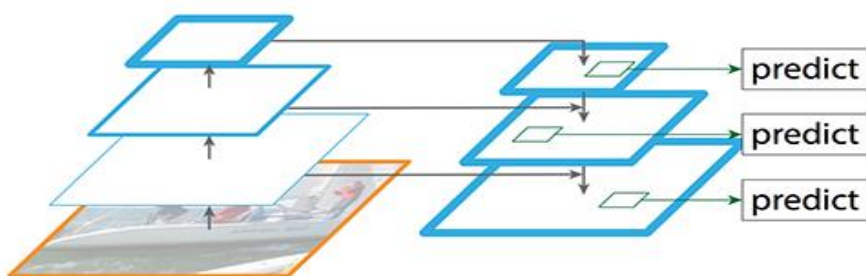


Figure 23:l'architecture FPN [49]

4.4.2 Modèle YOLO V5(2020) :

Les détecteurs d'objets basés sur CNN sont principalement applicables aux systèmes de recommandation. Les modèles YOLO (You Only Look Once) sont utilisés pour la détection d'objets haute performance.

YOLO divise l'image en un système de grille, et chaque grille détecte les objets en elle-même. Ils peuvent être utilisés pour découvrir des objets en temps réel sur la base de flux de données. Il nécessite très peu de ressources de calcul.

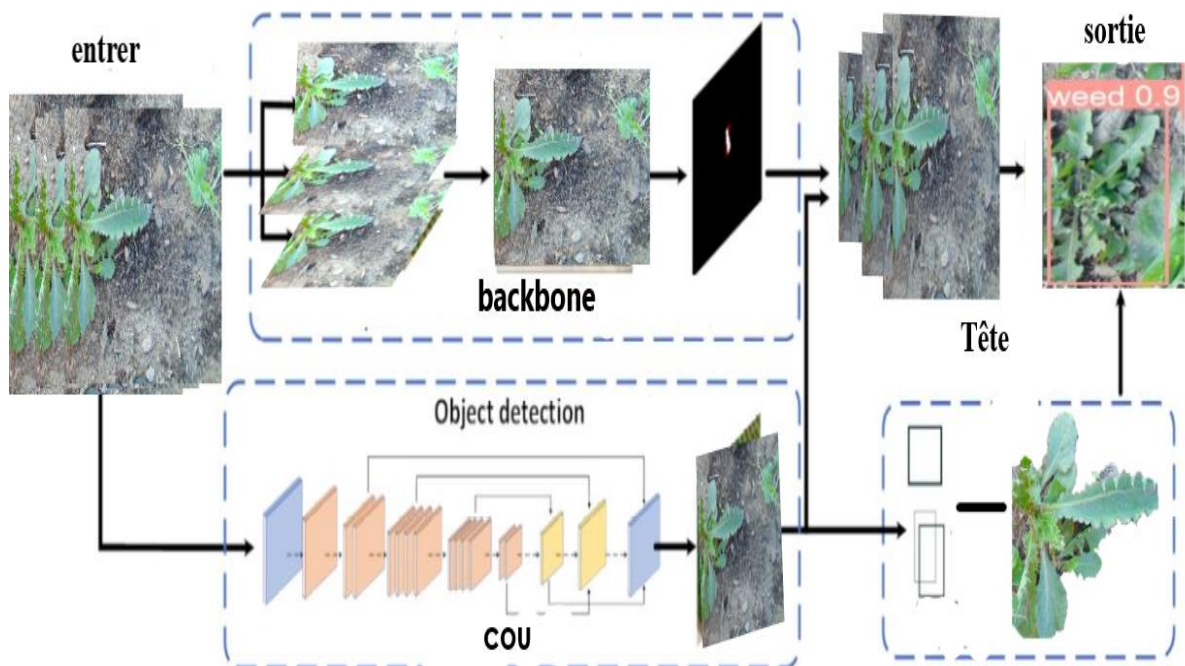


Figure 24:architecture de YOLOV5

L'architecture réseau de Yolov5. Il se compose de trois parties : (1) Backbone : CSPDarknet, (2) cou (Neck) : PANet et (3) Tête (Head) : Yolo Layer. Les données sont d'abord entrées dans CSPDarknet pour l'extraction de caractéristiques, puis transmises à PANet pour les caractéristiques la fusion.

➤ **Backbone :**

Le modèle de backbone est principalement utilisé pour extraire les caractéristiques clés de l'image d'entrée. CSP Darknet (Cross-Phase Partial Networks) est utilisé comme épine dorsale dans YOLO v5 pour extraire de riches propriétés utiles de l'image d'entrée.

➤ **Cou (neck) :**

Le modèle de cou est principalement utilisé pour créer des pyramides distinctives. Des modèles pyramidaux distinctifs aident à généraliser avec succès lorsqu'il s'agit de mesurer des objets. Il aide à reconnaître le même objet de différentes tailles et échelles.

Les pyramides distinguées sont très utiles pour aider les modèles à fonctionner efficacement sur des données inédites. D'autres modèles, tels que FPN, BiFPN et PANet, utilisent différents types d'approches de hiérarchie des fonctionnalités.

➤ **Tête (Head):**

La tête du modèle est principalement responsable de l'étape de détection finale. Utilise des boîtes liées pour créer des vecteurs de sortie finaux avec des probabilités de classe, des scores d'objet et des boîtes englobantes.

4.5 L'entraînement de modèle

L'entraînement de modèle est un ensemble de données utilisé pour former un algorithme ML. Il se compose des exemples de données de sortie et des ensembles correspondants de données d'entrée qui ont une influence sur la sortie.

Le modèle d'apprentissage est utilisé pour exécuter les données d'entrée dans l'algorithme afin de corrélérer la sortie traitée avec la sortie de l'échantillon.

4.6 Evaluation de modèle

Pour évaluer les modèles de détection d'objets tels que Mask R-CNN et YOLO, le taux d'erreur est utilisée ainsi que Le mAP compare la boîte englobante de la vérité au sol à la boîte détectée et renvoie un score. Plus le score est élevé, plus le modèle est précis dans ses détections.

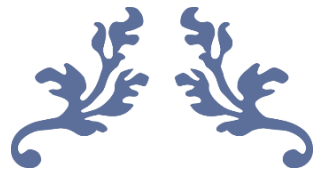
4.7 Prédiction et test des nouvelles images

Dans le chapitre suivant on va voir la prédiction du la base de données coté validation et le test de nouvelle images

5 Conclusion

Ce présent chapitre a été consacré aux la conception de nos système ainsi que architectures de nos modèles choisie 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 de Mask RCNN et yolov5 qui sont la méthodologie de notre recherche.

Dans le chapitre suivant nous avons listé les différents outils qui nous ont permis d'exécuter nos expérimentations ainsi que mis à la phase d'implémentation on détaillera comment nous avons mis en pratique ces technologies et exploité ces données afin de faire nos detection ainsi que leurs résultat.



Chapitre 05

Implémentation



1. Introduction

Après avoir vu la conception de notre système ainsi que les architectures de nos modèles choisies. Nous avons d'abord mis à la phase d'implémentation on détaillera les technologies exploitées dans notre recherche et comment nous avons mis en pratique ces technologies et exploité avec leur configuration de ces données afin de faire nos détecteurs ainsi que leurs résultats.

2. Les outils de développement utilisés

Dans cette partie nous présentons dans une première partie le langage de programmation Python utilisé dans l'implémentation de modèle et les outils logiciels tels que les bibliothèques : Keras, Tensorflow, Matplotlib, Numpy, Sklearn, tensorboard, OpenCV,... Et le matériel utilisé pour implémenter notre modèle.

2.1 Language de programmation



Python 3.7

Python est un langage de programmation interprété, multi-paradigme et multiplateformes. Il favorise la programmation impérative structurée, fonctionnelle et orientée objet. Il est conçu pour optimiser la productivité des programmeurs en offrant des outils de haut niveau et une syntaxe simple à utiliser [50]

2.2 Les Logiciels



Keras 2.1.0

Keras est une bibliothèque open source écrite en Python. La bibliothèque Keras permet d'interagir avec les algorithmes de réseaux de neurones profonds et d'apprentissage automatique. [51]



Tensorflow 1.15

Tensorflow est un outil open source d'apprentissage automatique développé par Google. Le code source a été ouvert le 9 novembre 2015 par Google et publié sous licence Apache [52].



NumPy 1.21.6

NumPy est une bibliothèque pour langage de programmation Python, destinée à manipuler des matrices ou tableaux multidimensionnels ainsi que des fonctions mathématiques opérant sur ces tableaux [53].



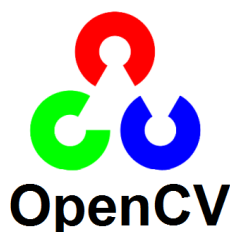
Matplotlib

Matplotlib est une bibliothèque du langage de programmation Python destinée à tracer et visualiser des données sous formes de graphiques. Elle peut être combinée avec les bibliothèques



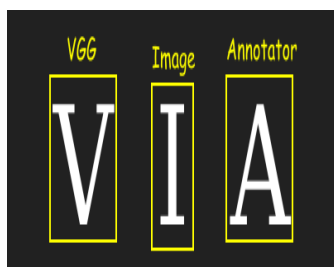
scikit-image

scikit-image est une collection d'algorithmes pour le traitement d'images. Il est disponible gratuitement et sans restriction [55].



OpenCV

OpenCV est une bibliothèque graphique libre, initialement développée par Intel, spécialisée dans le traitement d'images en temps réel. La société de robotique Willow Garage et la société ItSeez se sont succédé au support de cette bibliothèque [56].



VGG Image Annotator

VGG Image Annotator est un logiciel d'annotation manuelle simple et autonome pour l'image, l'audio et la vidéo. VIA s'exécute dans un navigateur Web et ne nécessite aucune installation ou configuration [57].



Roboflow

Roboflow est l'outil de conversion universel pour les ensembles de données de vision par ordinateur. Nous importons n'importe quel format d'annotation et exportons vers n'importe quel autre, ce qui signifie que vous pouvez passer plus de temps à expérimenter et moins de temps à lutter avec des scripts de conversion uniques [58].



Tensorboard

C'est un outil de mesure et de visualisation que nous avons utilisé pour l'amélioration des résultats, 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

3. Le matériel

Le Deep Learning est un domaine avec des exigences en calculs intenses et la disponibilité des Ressources (surtout en GPU) dédiés à cette tâche vont fondamentalement influencer sur L'expérience de l'utilisateur car sans ses ressources, il faudra trop de temps pour apprendre de ses erreurs ce qui peut être décourageant, alors nous avons choisi la plateforme Google Colaboratory pour effectuer les expérimentations.

Google Colab



Colaboratory, souvent raccourci en "Colab", permet à n'importe qui d'écrire et d'exécuter le code Python de son choix par le biais du navigateur. C'est un environnement particulièrement adapté à la machine Learning, à l'analyse de données et à l'éducation [60].

4. Simulation

Dans cette partie nous nous expliquons dans un petit tableau les paramètres de simulation de notre projet ainsi que les tailles d'images utilisées dans les deux modèles.

	Nombre des images	Taille d'image	Taux de train	Taux de detection	backbone
MASK-RCNN	200	858*644	3h	0.10s	Resnet101
YOLO-V5	200	858*644	46 min	0.014s	CSP Darknet

Tableau 3:Tableau de paramètre de simulation

5. Réalisation

Tant que nous travaillons sur la détection des mauvaises herbes à partir de deux modèles et que nous les comparons, l'application est divisée en deux parties, une partie sur la façon de travailler avec le masque RCNN et l'autre sur Yolo v5

5.1 Mask R-CNN

5.1.1 Connexion avec Colab

Premièrement nous Téléchargeons l'ensemble de données téléchargé sur Google Drive à partir du compte Gmail. Après cela, montez le lecteur Google sur le bloc-notes Colab comme illustré à la figure.

Nous Cliquons sur l'URL et sélectionnez le compte Gmail et entrons le code d'authentification.

```
[2] from google.colab import drive
drive.mount('/content/drive')

Mounted at /content/drive
```

Figure 25:connexion avec Colab

5.1.2 Importer les bibliothèques et le modèle MaskRCNN

Nous avons utilisé le modèle prédéfini Mask RCNN on applique des petites modifications sur les paramètres pour obtenir les besoins de notre but.

```

1 import os
2 import sys
3 import json
4 import datetime
5 import numpy as np
6 import skimage.draw
7 import cv2
8 from mrcnn.visualize import display_instances
9 import matplotlib.pyplot as plt
10
11 # Root directory of the project
12 ROOT_DIR = os.path.abspath("../..")
13
14 # Import Mask RCNN
15 sys.path.append(ROOT_DIR) # To find local version of the library
16 from mrcnn.config import Config
17 from mrcnn import model as modellib, utils

```

Figure 26: Import des bibliographies

La configuration appliquée sur notre modèle de detection

```

30 class CustomConfig(Config):
31     """Configuration du training on the weed dataset.
32     """
33     NAME = "weed"
34
35     # GPU with 12GB memory, can fit two images.
36     IMAGES_PER_GPU = 2
37
38     # Number classes (avec background)
39     NUM_CLASSES = 1 + 1 # Background + weed
40
41     # Number steps per epoch
42     STEPS_PER_EPOCH = 100
43
44     # Skip detections with < 90% confidence
45     DETECTION_MIN_CONFIDENCE = 0.9
46
47

```

Figure 27: Configuration de l'entraînement des mauvaises herbes

```
[ ] config = weed.CustomConfig()
    WEED_DIR = "/content/dataset"

[ ] # Ouvrir du configurations du model
    # changes du training a inferencing.
    class InferenceConfig(config.__class__):
        # Run du 1 image dans temps reel
        GPU_COUNT = 1
        IMAGES_PER_GPU = 1

    config = InferenceConfig()
    config.display()
```

Figure 28:la configuration de modèle

```
[ ] Configurations:
    BACKBONE                resnet101
    BACKBONE_STRIDES        [4, 8, 16, 32, 64]
    BATCH_SIZE              1
    BBOX_STD_DEV            [0.1 0.1 0.2 0.2]
    COMPUTE_BACKBONE_SHAPE  None
    DETECTION_MAX_INSTANCES 100
    DETECTION_MIN_CONFIDENCE 0.9
    DETECTION_NMS_THRESHOLD 0.3
    FPN_CLASSIF_FC_LAYERS_SIZE 1024
    GPU_COUNT               1
    GRADIENT_CLIP_NORM      5.0
    IMAGES_PER_GPU          1
    IMAGE_CHANNEL_COUNT      3
    IMAGE_MAX_DIM           1024
    IMAGE_META_SIZE         14
    IMAGE_MIN_DIM           800
    IMAGE_MIN_SCALE         0
    IMAGE_RESIZE_MODE        square
    IMAGE_SHAPE              [1024 1024  3]
    LEARNING_MOMENTUM        0.9
    LEARNING_RATE           0.001
    LOSS_WEIGHTS             {'rpn_class_loss': 1.0, 'rpn_bbox_loss': 1.0, 'mrcnn_class_loss': 1.0, 'mrcnn_bbox_loss': 1.0, 'mrcnn_ma
    MASK_POOL_SIZE          14
    MASK_SHAPE              [28, 28]
    MAX_GT_INSTANCES        100
    MEAN_PIXEL              [123.7 116.8 103.9]
    MINI_MASK_SHAPE         (56, 56)
    NAME                    weed
    NUM_CLASSES              2
    POOL_SIZE               7
```

Figure 29:l'affichage de configuration total des modèles utilisés

5.1.3 Accès aux données

La base de données contient de partie une partie de train et partie de validation (test) le schéma dessus exprime ca :

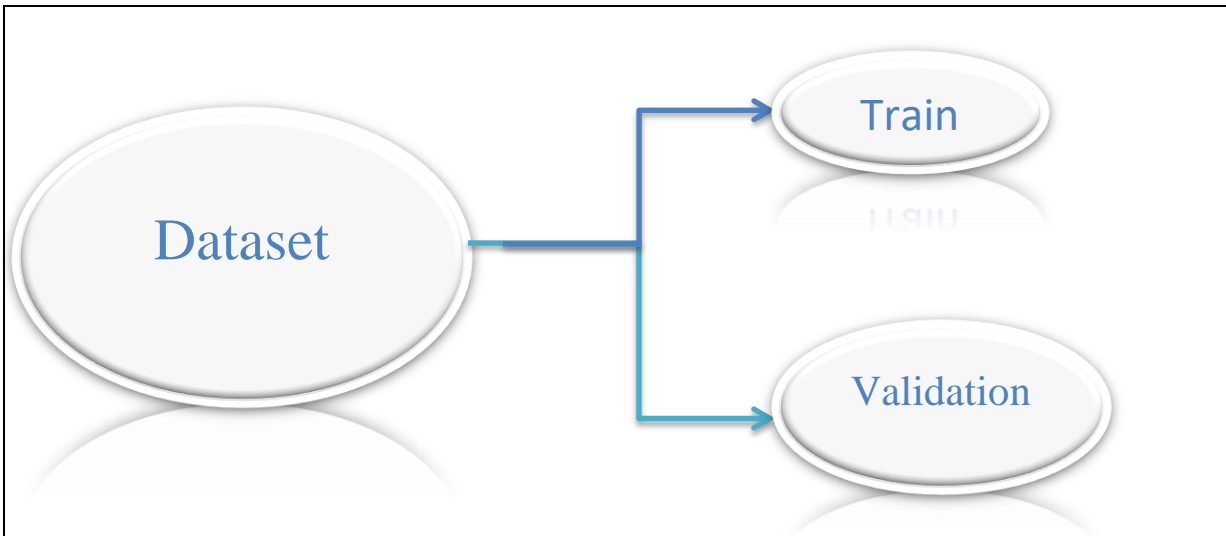


Figure 30:schéma de base de données

```
[ ] !unzip -q '/content/drive/MyDrive/weed_project_detection/dataset.zip' # unzip the dataset
```

Figure 31:Unzip de donnée

```

59 class CustomDataset(utils.Dataset):
60     def load_custom(self, dataset_dir, subset):
61         """prend a subset de weed dataset.
62         Root directory dataset.
63         subset: Subset to load: train or val
64         """
65         # Add classes. on a seule class .
66         self.add_class("weed", 1, "weed")
67
68         # Train or validation ?
69         assert subset in ["train", "val"]
70         dataset_dir = os.path.join(dataset_dir, subset)
71
72         annotations1 = json.load(open(os.path.join(dataset_dir, "via_region_data.json")))
73         annotations = list(annotations1.values())
74         # L'outil VIA enregistre les images dans le JSON même si elles n'en ont pas
75         # annotations. Ignorer les images non annotées.
76         annotations = [a for a in annotations if a['regions']]
77         # Add images
78         for a in annotations:
79             # Obtenir les coordonnées x, y des points des polygones qui composent
80             # le contour de chaque instance d'objet. Il y a des magasins dans le
81             # shape_attributes format json ci-dessus
82             polygones = [r['shape_attributes'] for r in a['regions'].values()]
83
84             # load_mask() a besoin de la taille de l'image pour convertir les polygones en masques.
85             # Malheureusement, VIA ne l'inclut pas dans JSON, nous devons donc lire
86             image_path = os.path.join(dataset_dir, a['filename'])
87             image = skimage.io.imread(image_path)
88             height, width = image.shape[:2]
89
90             self.add_image(
91                 "weed", ## for a single class just add the name here
92                 image_id=a['filename'], # use file name as a unique image id
93                 path=image_path,
94                 width=width, height=height,
95                 polygones=polygones)
  
```

Figure 32:Accès aux données

5.1.4 L'entraînement du modèle

Comme nous utilisons une base de données petit (200 images), nous partons de Poids entraînés COCO, nous n'avons pas besoin de nous entraîner longtemps.

```

132 def train(model):
133     """Train the model."""
134     # Training dataset.
135     dataset_train = CustomDataset()
136     dataset_train.load_custom(args.dataset, "train")
137     dataset_train.prepare()
138
139     # Validation dataset
140     dataset_val = CustomDataset()
141     dataset_val.load_custom(args.dataset, "val")
142     dataset_val.prepare()
143
144     print("Training network ")
145     model.train(dataset_train, dataset_val,
146               learning_rate=config.LEARNING_RATE,
147               epochs=40,
148               layers='all')

```

Figure 33: la fonction de l'entraînement

```

!python3 weed.py train --dataset=/content/dataset --weights=coco

```

Deprecated in NumPy 1.20; for more details and guidance: <https://numpy.org/devdocs/release/1.20.0-notes.html#deprecations>
return mask.astype(np.bool), np.ones([mask.shape[-1]], dtype=np.int32)
95/100 [=====>..] - ETA: 7s - loss: 0.2054 - rpn_class_loss: 0.0011 - rpn_bbox_loss: 0.0160 - mrcnn_class_loss: 0.013
Deprecated in NumPy 1.20; for more details and guidance: <https://numpy.org/devdocs/release/1.20.0-notes.html#deprecations>
return mask.astype(np.bool), np.ones([mask.shape[-1]], dtype=np.int32)
weed.py:121: DeprecationWarning: `np.bool` is a deprecated alias for the builtin `bool`. To silence this warning, use `bool` by itself. Do
Deprecated in NumPy 1.20; for more details and guidance: <https://numpy.org/devdocs/release/1.20.0-notes.html#deprecations>
return mask.astype(np.bool), np.ones([mask.shape[-1]], dtype=np.int32)
96/100 [=====>..] - ETA: 6s - loss: 0.2047 - rpn_class_loss: 0.0011 - rpn_bbox_loss: 0.0159 - mrcnn_class_loss: 0.013

Figure 34: L'exécution de train

5.1.5 Validation de la base de données

Nous validons par prédiction nos images de validation 19 images de validation

```

# validation dataset
dataset = weed.CustomDataset()
dataset.load_custom(WEED_DIR, "val")

# using the dataset
dataset.prepare()

print("Images: {}\nClasses: {}".format(len(dataset.image_ids), dataset.class_names))

```

Images: 19
Classes: ['BG', 'weed']

Figure 35: Le contenu de validation

5.1.6 Prédiction

```

▶ image_id = random.choice(dataset.image_ids) # run d'images de dataset validation
image, image_meta, gt_class_id, gt_bbox, gt_mask = \
modellib.load_image_gt(dataset, config, image_id, use_mini_mask=False)
info = dataset.image_info[image_id]
print("image ID: {}.{} ({}).{}".format(info["source"], info["id"], image_id,
dataset.image_reference(image_id)))

# Run object detection
results = model.detect([image], verbose=1)

# Display results
ax = get_ax(1)
r = results[0]
visualize.display_instances(image, r['rois'], r['masks'], r['class_ids'],
dataset.class_names, r['scores'], ax=ax,
title="Predictions")

log("gt_class_id", gt_class_id)
log("gt_bbox", gt_bbox)
log("gt_mask", gt_mask)

image ID: weed.196.jpg (13) /content/dataset/val/196.jpg
Processing 1 images
image                shape: (1024, 1024, 3)    min: 0.00000    max: 255.00000    uint8
molded_images        shape: (1, 1024, 1024, 3) min: -123.70000   max: 151.10000   float64
image_metas          shape: (1, 14)           min: 0.00000     max: 1024.00000  int64
anchors              shape: (1, 261888, 4)    min: -0.35390    max: 1.29134     float32
gt_class_id          shape: (1,)              min: 1.00000     max: 1.00000     int32
gt_bbox              shape: (1, 4)            min: 128.00000   max: 895.00000   int32
gt_mask              shape: (1024, 1024, 1)  min: 0.00000     max: 1.00000     bool
    
```

Figure 36:La fonction de prédiction



Figure 37:prédiction images de validation

5.2 Yolov5 :

Le travaille avec le yolov5 est plus facile que Mask RCNN on prend la même base de donnée et annoter avec le roboflow et exporter de format yolov5 pytorch

5.2.1 Importer la bibliothèque et le modèle

```

# install dependencies as necessary
!pip install -qr requirements.txt # install dependencies (ignore errors)
import torch

from IPython.display import Image, clear_output # to display images
from utils.google_utils import gdrive_download # to download models/datasets

# clear_output()
print('Setup complete. Using torch %s %s' % (torch.__version__, torch.cuda.get_device_properties(0) if torch.cuda.is_available() else 'CPU'))

596 kB 6.9 MB/s

#follow the link below to get your download code from from Roboflow
!pip install -q roboflow
from roboflow import Roboflow
rf = Roboflow(model_format="yolov5", notebook="roboflow-yolov5")

145 kB 14.6 MB/s
178 kB 64.5 MB/s
1.1 MB 73.2 MB/s
67 kB 6.1 MB/s
54 kB 3.4 MB/s
138 kB 69.2 MB/s
63 kB 608 kB/s
Building wheel for roboflow (setup.py) ... done
Building wheel for wget (setup.py) ... done
ERROR: pip's dependency resolver does not currently take into account all the packages that are installed. This behaviour is the source of the following dependency con
google-colab 1.0.0 requires requests<=2.23.0, but you have requests 2.27.1 which is incompatible.
datascience 0.10.6 requires folium==0.2.1, but you have folium 0.8.3 which is incompatible.
albumations 0.1.12 requires imgaug<0.2.7,>=0.2.5, but you have imgaug 0.2.9 which is incompatible.
upload and label your dataset, and get an API KEY here: https://app.roboflow.com/?model=yolov5&ref=roboflow-yolov5

```

Figure 38: les bibliothèques importer

5.2.2 Accédé aux données

```

! cd /content/yolov5

from roboflow import Roboflow
rf = Roboflow(api_key="Zp9pUWmQuYk146cp6mpr")
project = rf.workspace("jihene-lakehal").project("weedss")
dataset = project.version(1).download("yolov5")

/content/yolov5
loading Roboflow workspace...
loading Roboflow project...
Downloading Dataset Version Zip in weedss-1 to yolov5pytorch: 100% [9888292 / 9888292] bytes
Extracting Dataset Version Zip to weedss-1 in yolov5pytorch:: 100% [403/403 [00:00<00:00, 1517.64it/s]

# this is the YAML file Roboflow wrote for us that we're loading into this notebook with our data
!cat (dataset.location)/data.yaml

names:
- weed
nc: 1
train: weedss-1/train/images
val: weedss-1/valid/images

```

Figure 39: accès aux données

5.2.3 L'entrainement du modèle

```
[ ] # train yolov5s on custom data for 100 epochs
# time its performance
%%time
%cd /content/yolov5/
!python train.py --img 416 --batch 16 --epochs 100 --data (dataset.location)/data.yaml --cfg ./models/custom_yolov5s.yaml --weights '' --name yolov5s_results --cache
```

Figure 40:l'entrainement du modèle yolov5

5.2.4 Validation

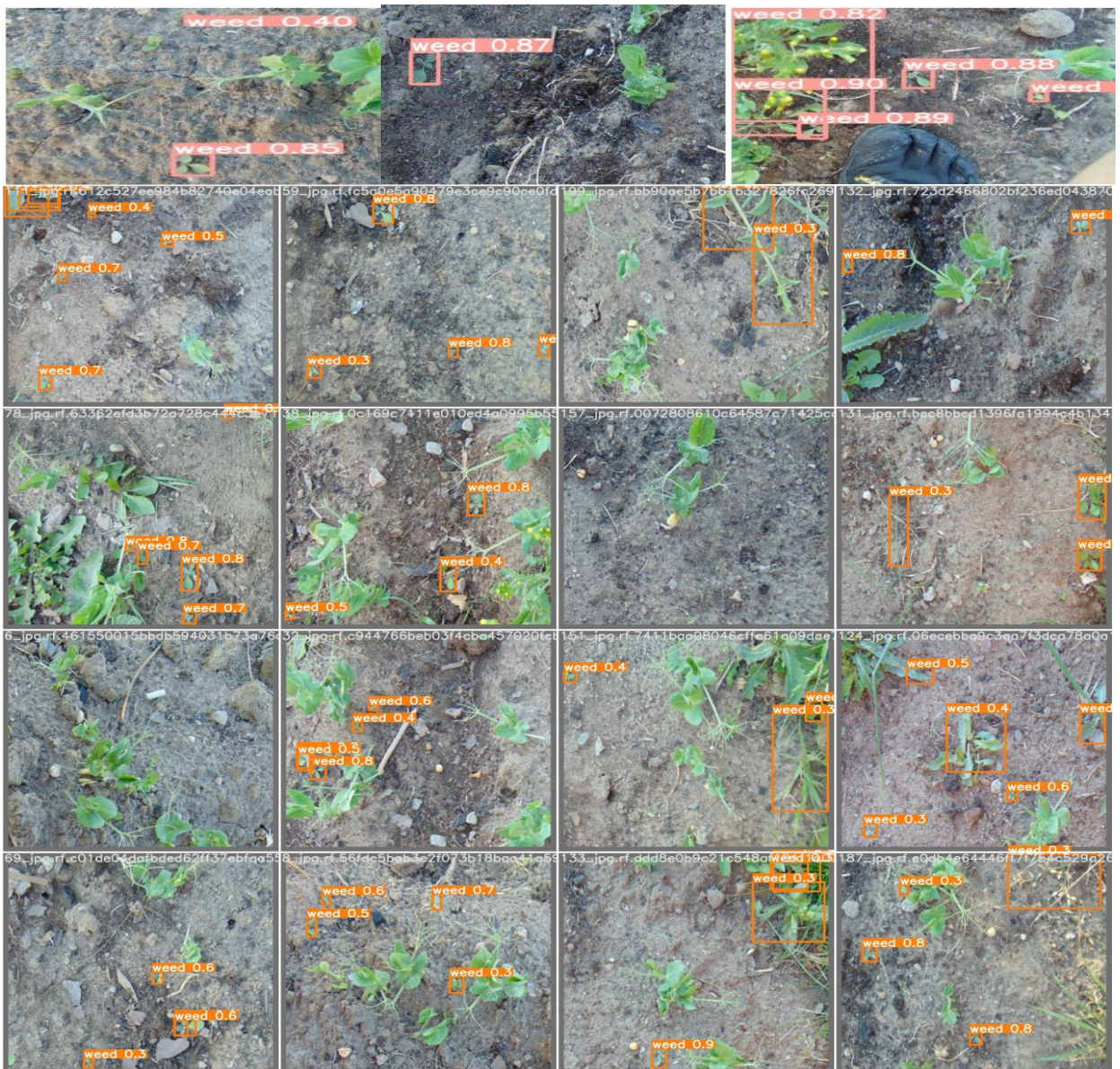


Figure 41:validation des images

6. Résultats et discussion

Afin de montrer les résultats obtenus pour nos modèles, on illustre dans ce qui suit les résultats en termes d'erreur ainsi que la comparaison entre les deux modèles.

6.1 Taux d'erreur (loss)

Pour notre modèle, nous avons commencé avec le modèle MASK RCNN avec notre ensemble de données et le nombre d'époques 40.

Après analyse des résultats obtenus, nous faisons les observations suivantes : Selon les nombres, le taux d'erreur d'apprentissage et de validation diminue avec le nombre de périodes jusqu'à stabilité, il est donc normal que le taux d'apprentissage et de validation augmente avec le nombre d'époques et cela reflète qu'à chaque époque apprend Form plus d'informations.

Au cours de l'entraînement , plusieurs métriques de RPN [rpn_class, rpn_bbox] et de zone de retour sur investissement (rpn_rois) obtenues par RPN, [output_rois] obtenues par rpn_rois et l'étiquette au sein de la suite ont été obtenues cible et masque les résultats de détection des deux sous-classes maskrcnn [loss, mrcnn_class, mrcnn_bbox, mrcnn_mask]

En plus des métriques de perte [rpn_class_loss , rpn_bbox_loss , class_loss , bbox_loss , mask_loss]

Et dans la prédiction de sortie du réseau : basée sur le résultat de détection obtenu sur la base de la région d'intérêt rpn_rois obtenue par le réseau RPN, [rpn_rois, rpn_class, rpn_bbox] et [mrcnn_class, mrcnn_bbox] obtenus par rpn_rois et la découverte de l'entrée de caractéristique de branche mask, (mrcnitsn_class est également disponible mais pas la sortie) De plus, les résultats de la branche de détection king et maskrcnn obtenus par rpn obtiennent le résultat de détection final [detection] Enfin, la branche de détection et maskrcnn du masque de détection, [mrcnn_mask]



Figure 42:le taux d'erreur (loss) de modèle Mask RCNN du 40 épochs



Figure 43:Le taux d'erreur de class



Figure 44:Le taux d'erreur de cadre

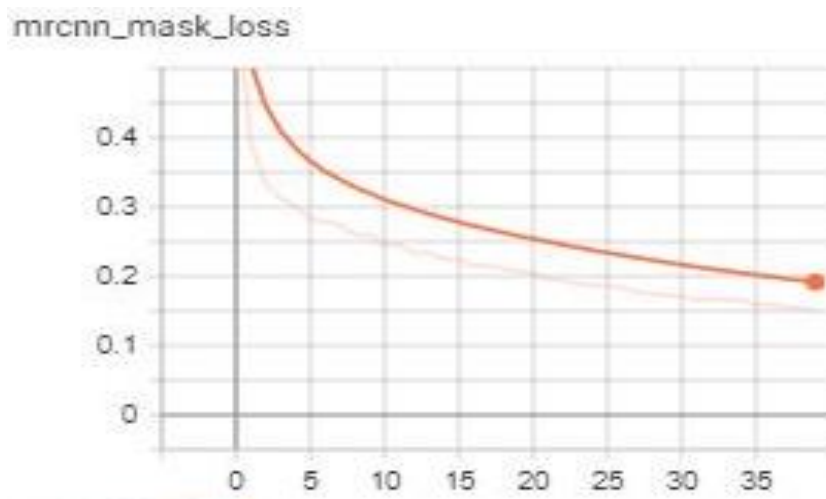


Figure 45:Le taux d'erreur de masque



Figure 46:Le taux d'erreur de class de la région



Figure 47:Le taux d'erreur de cadre de la région

Nous avons utilisé un ensemble de données composé de 202 images dans lesquelles 183 images ont été utilisées pour la formation et 19 images pour la validation. Il a fallu environ 80 minutes pour 40 époques, chaque période contenant 100 (batch) impulsions d'entraînement.

Un taux de perte d'environ 0,18 alors Notre modèle toujours une précision décente malgré notre petit ensemble de données d'entraînement. Tensorboard est utilisé pour la visualisation des données

En suite en va voir le modèle YoloV5

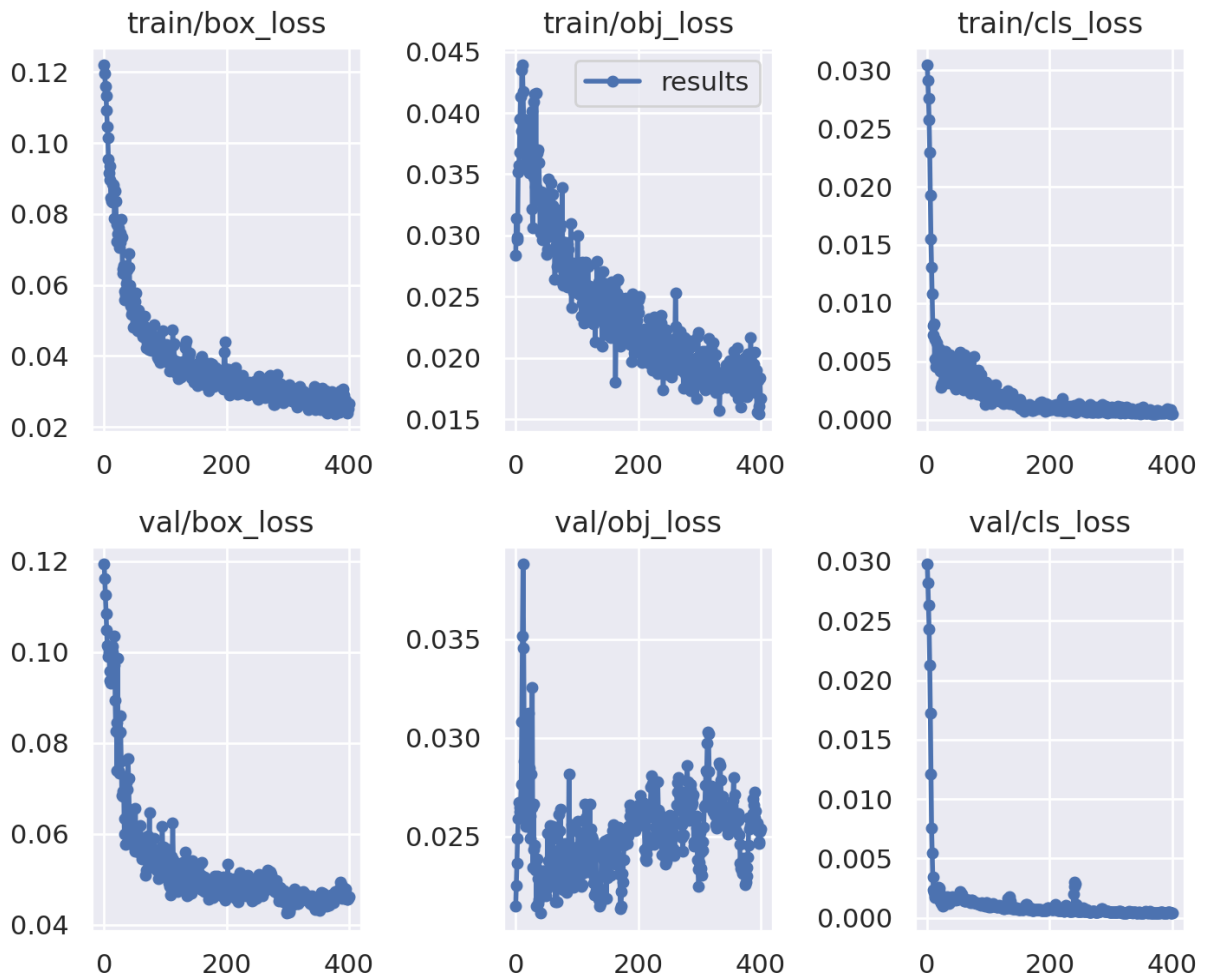


Figure 48: Les résultats obtenus à partir du modèle YoloV5

La fonction de perte YOLO est composée de trois parties :

- **Box_loss** : perte de régression de la boîte englobante (erreur quadratique moyenne).
- **Obj_loss** : la confiance de la présence de l'objet est la perte d'objectivité (Binary Cross Entropy).
- **Cls_loss** : la perte de classification (Cross Entropy).

Étant donné que nos données n'ont que classe (Weed) et background, l'erreur d'identification de classe et l'erreur de classification est d'environ 0.000.

6.2 Rapport d'évaluation de detection

Pour évaluer les performances d'un modèle de façon complète, vous devez analyser à la fois la précision et le rappel Pour le Mask RCNN Le mAP compare la boîte englobante de la

vérité au sol à la boîte détectée et renvoie un score. Plus le score est élevé, plus le modèle est précis dans ses détections.

```

# Compute VOC-Style mAP @ IoU=0.5
# Running on 10 images. Increase for better accuracy.
image_ids = np.random.choice(dataset.image_ids, 2)
APs = []
for image_id in image_ids:
    # Load image and ground truth data
    image, image_meta, gt_class_id, gt_bbox, gt_mask = \
        modellib.load_image_gt(dataset, config,
                                image_id, use_mini_mask=False)
    molded_images = np.expand_dims(modellib.mold_image(image, config), 1)
    # Run object detection
    results = model.detect([image], verbose=0)
    r = results[0]
    # Compute AP
    AP, precisions, recalls, overlaps = \
        utils.compute_ap(gt_bbox, gt_class_id, gt_mask,
                        r["rois"], r["class_ids"], r["scores"], r["masks"])
    APs.append(AP)

print("mAP: ", np.mean(APs))

```

mAP: 0.7777777860562006

Figure 49:évaluation de detection mAP avec Mask RCNN

```

[ ] # Draw precision-recall curve
AP, precisions, recalls, overlaps = utils.compute_ap(gt_bbox, gt_class_id, gt_mask,
                                                    r['rois'], r['class_ids'], r['scores'], r['masks'])
visualize.plot_precision_recall(AP, precisions, recalls)

```

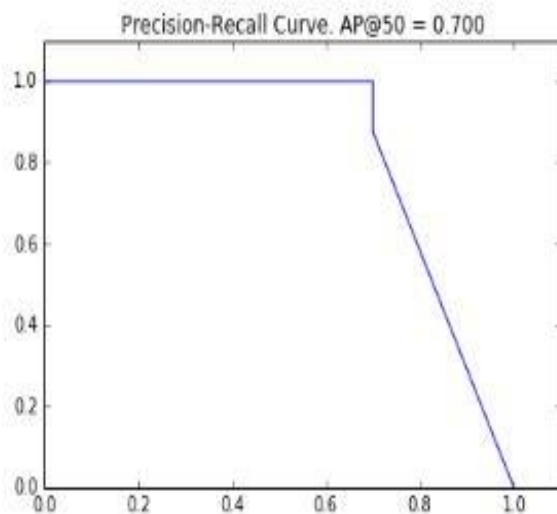


Figure 50:évaluation de précision de Mask R-CNN

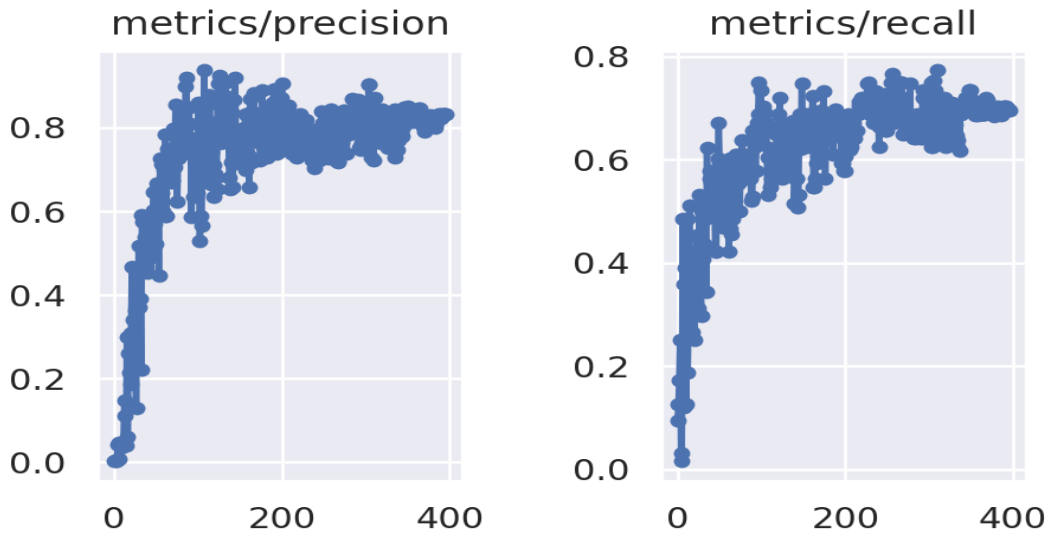


Figure 51:évaluation de détection avec YOLO

7. Comparaison

Dans ce tableau, les métriques obtenues par nos modèles sont présentées :

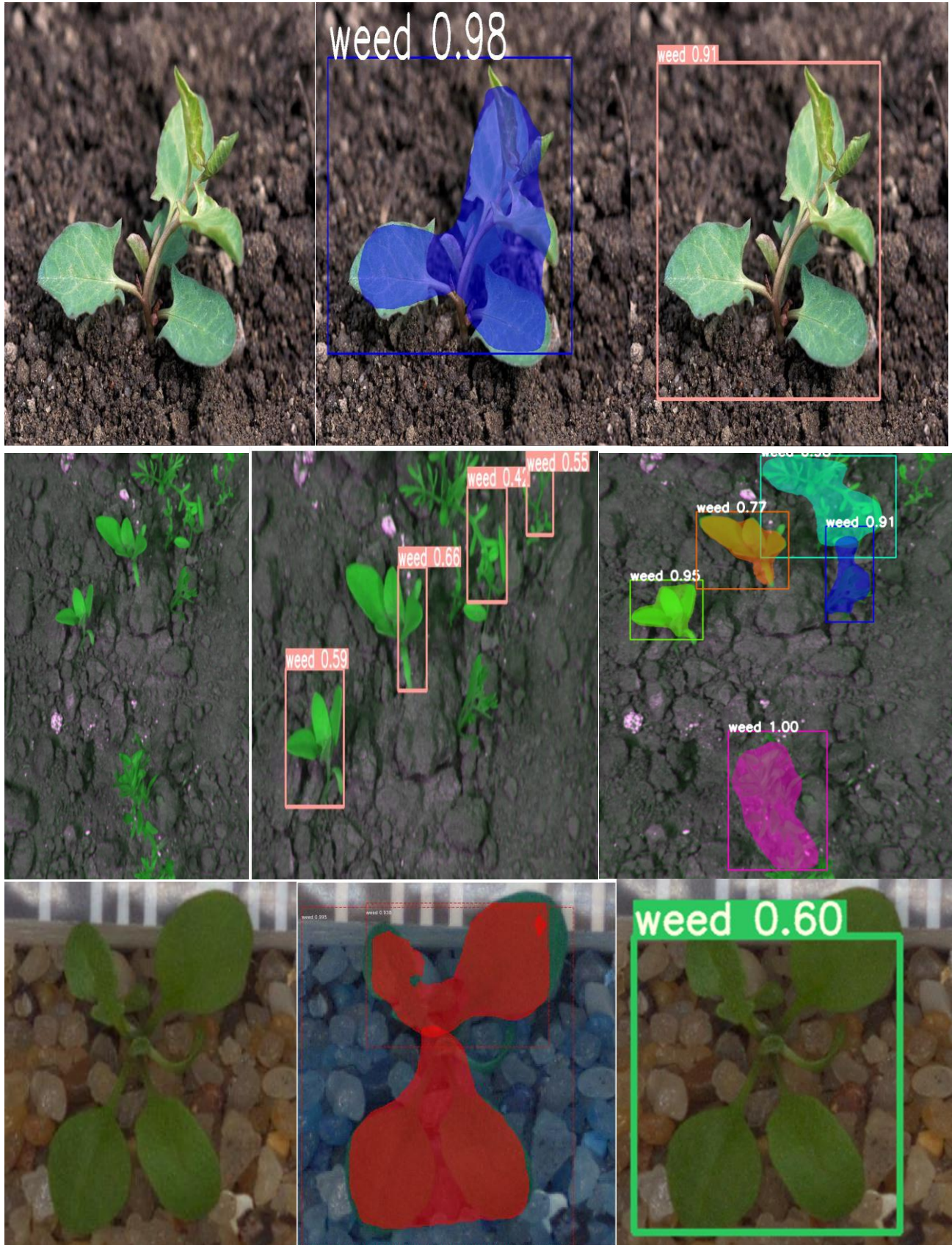
	Taux d'erreur	Taux d'erreur de classe	Taux d'erreur de box	Précision	Précision-recall
MASK-RCNN	0.19	0.02	0.03	0.777	0.700
YOLO V5	0.3	0.000	0.04	0.89	0.77

Tableau 4: tableaux de comparaison

La comparaison des performances de Mask R-CNN et YOLOv5 vise à produire les meilleurs modèles de détection et de reconnaissance des mauvaises herbes. Sur la base d'expériences, YOLOv5 a surpassé Mask R-CNN avec un score de précision de 0,89 et des temps d'entraînement plus rapides. Cependant, la fonction de Mask R-CNN ajoute un masque à la détection finale, ce qui montre une meilleure segmentation des motifs avec un masque ainsi que le taux de reconnaissance est mieux que le yolov5.

8. Test de nouvelle image

La figure suivant illustre le résultat de 10 enchainant de test des images des différentes bases de données.



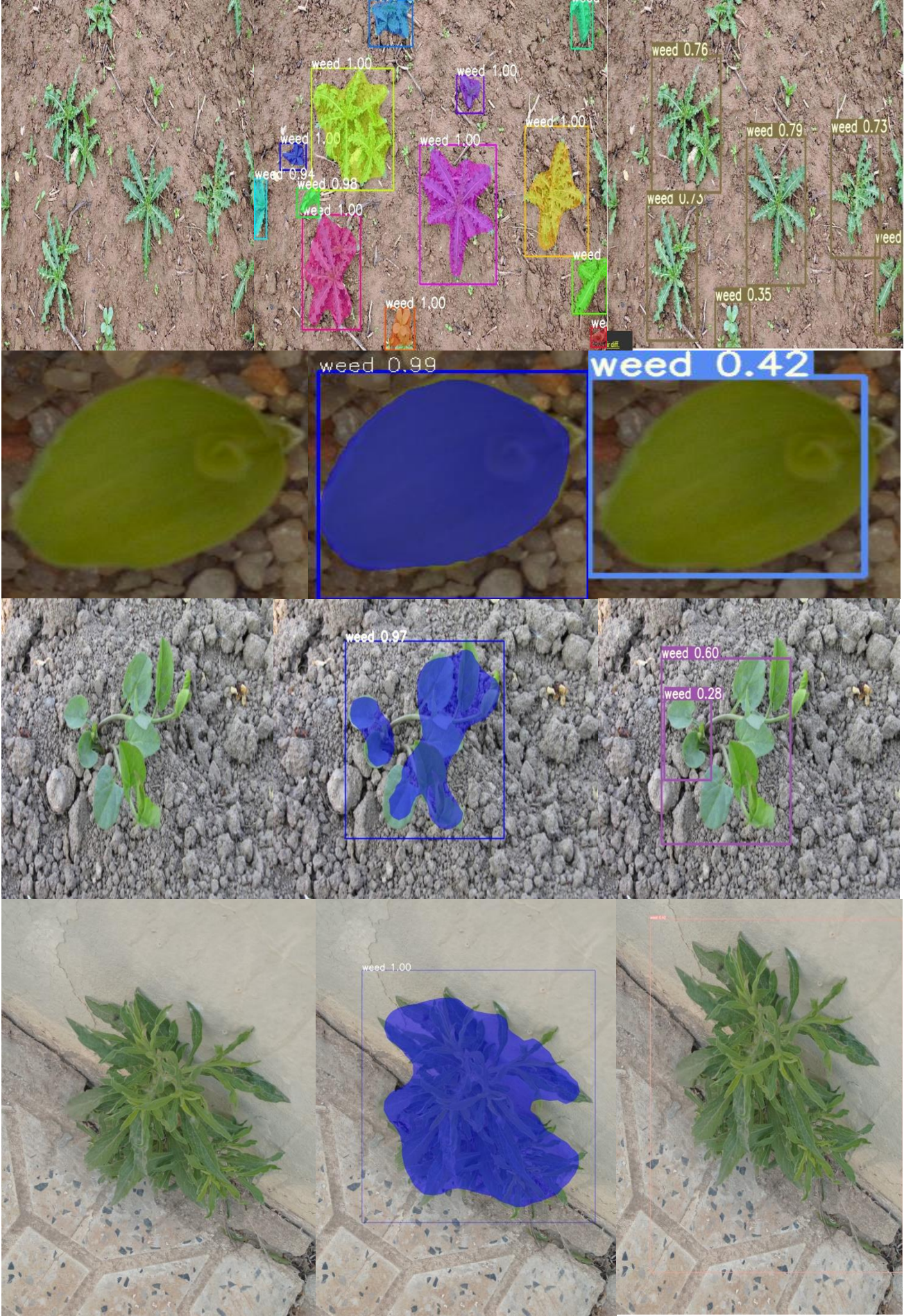




Figure 52: Test des nouvelles images

9. Discussion

L'aspect visuel contour de mauvaises herbes peut être utilisé pour identifier et le détecter. Un procédé de segmentation qui peut bien isoler ces certaines parties d'une image est nécessaire pour une analyse plus approfondie dans un système de détection automatique de mauvaises herbes existé.

Dans cette recherche, nous avons étudié les performances de deux méthodes de segmentation basées sur CNN, à savoir YOLO et Mask R-CNN, pour le détecter. Nous avons recyclé les modèles préformés YOLO et Mask R-CNN sur l'ensemble de données Weed composé d'images avec une forte variabilité de l'arrière-plan, de l'éclairage.

L'expérience sur 200 images annotées manuellement a indiqué que les deux modèles fonctionnent de manière optimale. Les performances de YOLO étaient légèrement meilleures que celles de Mask R-CNN, avec un taux d'erreur de 3 % ainsi que 19% pour Mask R-CNN mais le MaskRCNN est plus précis dans la prédiction que Yolo car le Mask RCNN définit le Mask plus claire.

Nous avons montré échantillons de tailles différentes des bases de données disposée en ligne, qu'il est possible d'obtenir un bon résultat n'importe où.

10. Conclusion

Dans ce chapitre, nous introduisons une approche d'apprentissage en profondeur basée sur un réseau de neurones Convolutif (CNN) pour la détection des mauvaises herbes. Notre système de détection de mauvaises herbes proposé est un masque RCNN et un modèle basé sur yolov5 mis en œuvre à l'aide de certaines bibliothèques Google Colab telles que Keras, Numpy, etc.

Pour s'entraîner sur la base de données d'herbes, qui comprend 200 images et tester sur de nouvelles images du nouveau jeu de données, nous testons sur plusieurs échantillons.

Il s'avère que notre modèle est un succès, car des expériences sur 200 images annotées manuellement montrent que les deux modèles fonctionnent le mieux.

YOLOv5 fonctionne légèrement mieux que Mask R-CNN avec 2% et 19% de taux d'erreur Mask R-CNN, mais les prédictions de MaskRCNN sont plus précises que celles de Yolo car Mask R-CNN définit l'objectif plus clairement et plus précisément.



Conclusion général et perspectives



Conclusion générale

Cette recherche s'inscrit dans le cadre d'une contribution basée sur des algorithmes d'intelligence artificielle pour aider à la détection des mauvaises herbes qui a un impact profond sur la production agricole ainsi que la production alimentaire des populations car elles diminuassent à cause de ces derniers.

Le traitement de données (machine Learning ou Deep Learning) connaît un grand essor il impacte plusieurs domaines et spécialités dans la recherche ou le monde professionnel. Le domaine agricole n'échappe pas à cette réalité notamment avec l'agriculture intelligente de précision. Ainsi l'utilisation des nouvelles techniques est au sein du processus de bout en bout, de façon directe ou indirecte. Le DL n'est qu'un moyen potentiellement efficace d'offrir des solutions prometteuses basées sur celui-ci sur les données pour aider les agriculteurs à faire face aux mauvaises herbes.

Notre recherche se focalisera sur un des problèmes majeurs de l'agriculture qui est la prolifération des mauvaises herbes dans les champs agricoles, La détection préalable et mieux ciblée des mauvaises herbes pourra jouer un grand rôle pour la promotion d'une agriculture intelligente, car cela permettra une grande diminution de l'utilisation des herbicides et produits chimiques dans le cas d'une culture traditionnelle ou alors l'arracher manuellement ou automatiser, les méthodes, ensembles de données, outils et performances DL sont disponibles en fonction de l'intelligence artificielle.

Nous avons commencé par fournir un aperçu complet des mauvaises herbes et de ses types existants, Sa prévalence. Ensuite, nous avons fait une étude détaillée sur les dernières méthodologies et applications actuelles des chercheurs DL.

Nous avons présenté un modèle détaillé basé sur deux algorithmes plus connus yoloV5 et MaskRCNN pour construire un système automatique plus précis notre Ensemble de données des mauvaises herbes dans la province de Tébessa. Malgré la petite taille du jeu de données utilisé, les résultats obtenus sont très encourageants.

Perspectives

Nous n'avons travaillé que dans sa version initiale, et il est toujours ouvert aux améliorations et à la comparaison du travail avec d'autres. Le modèle peut être encore amélioré en collectant plus d'ensembles de données pour différentes espèces de mauvaises herbes, en augmentant le nombre d'itérations de formation et en augmentant les images, aidant à mettre en œuvre la même chose dans le champ agricole réel à l'aide de drones et de techniques de pulvérisation robotisées.

Il peut également être utilisé pour détecter tout type particulier de mauvaise herbe en un seul clic sur une image en utilisant le même modèle, une version allégée des applications pour smartphone. Il peut aider les producteurs à détecter et identifier rapidement et clairement les mauvaises herbes.

Listes des Références

- [1]. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, A survey on sensor networks. IEEE Communications Magazine, vol 40, pp. 102-114, August, 2002, wsn. (s. d.). Wsn. Consulté le 2022, à l'adresse <https://www.technoscience.net/illustration/Definition/inconnu/w/WSN.svg>
- [2]. Challal, Y. A. C. I. N. E. (Éd.). (2008, novembre). Réseaux de Capteurs Sans Fils (No version). <http://creativecommons.org/licenses/by-nc/2.0/fr/>
- [3]. CAYIRCI, E. (2004). "Wireless sensor networks". In : D. Katsaros et al. (éd), Wireless information highways (pp. 273-301). Hershey : Idea group inc.
- [4]. Ian F Akyildiz, Weilian Su, Yogesh Sankarasubramaniam, and Erdal Cayirci. A survey on sensor networks. Communications magazine, IEEE, 40(8) :102–114,2002
- [5]. Ibtissem Boulanouar. Algorithmes de suivi de cible mobile pour les réseaux de capteurs sans fils. Computer Science. Université Paris-Est, 2014. French. <NNT : 2014PEST1077>. <tel01124323>
- [6]. Philippe Zwaenepoel & Jean-Michel Le Bars (1977), L'agriculture de précision ; Ingénieries EAT no 12, décembre 1997, p. 67 à 79
- [7]. Tissut, Delval, Mamarot et Ravanel, 2006.Plantes, herbicides et désherbage. Association de Coordination Technique Agricole, Paris : page 481-588
- [8]. Roger., 2013Les mauvaises herbes agricoles. Edition Berger A.C. inc.Canada,
- [9]. Mauvaise herbe — Les Mots de l'agronomie. (s. d.). Mauvaise herbe — Les Mots de l'agronomie. Consulté le 9 février 2022, à l'adresse https://mots-agronomie.inra.fr/index.php/Mauvaise_herbe
- [10]. LES MAUVAISES HERBES DES AGRUMES DE LA REGION DE MOSTAGANEM.(2019,juillet).<http://e-biblio.univmosta.dz/bitstream/handle/123456789/13044/rihab.pdf?sequence=1&isAllowed=y>
- [11]. Maillet J., 1981 – Evolution de la flore adventice dans le montpelliérais sous la pression des techniques culturales, thèse de docteur-ingénieur, biologie et ecologie végétales,USTL, montpellier, France. Montpellier, USTL, 200 p
- [12]. Maillet J., 1991- Constitution et dynamique des communautés de mauvaises herbes de France et des rizières de camargue. Th. Doc. Etat. Univ. Montpellier, 163 p.
- [13]. Bio, A. P. (2022, 28 février). Les 4 avantages des mauvaises herbes au potager. Au potager bio - Conseils pour votre potager biologique. <https://au-potager-bio.com/les-4-avantages-des-mauvaises-herbes-au-potager/>

- [14]. Contrôle de mauvaises herbes. (2017, 25 janvier). controle des mauvaises herbes. Consulté le 2022, à l'adresse <https://www.mapaq.gouv.qc.ca/SiteCollectionDocuments/Regions/LavalLanaudiere/Jurneesagricoles2017/Controleledesmauvaisesherbes.pdf>
- [15]. BAILLY R., AGUILAR J., FAIVRE-AMIOT A., MIMAUD J., PAITIER G., CASSEDANNEP. ; CHOPPIN DE JANVRY E., LE NAIL F., 1980 - guide pratique de défense des cultures Ed. ACTA paris 74 p 420 p.
- [16]. Hanitet Karima. (2011–2012). Les GROUPEMENTS des ADVENTICES des CULTURES dans la REGION d'ORAN. univ oran. <https://theses.univ-oran1.dz/document/TH3918.pdf>
- [17]. SBIKI, M. (2015–2016, juin). Contribution à l'étude du régime alimentaire et de la biologie de reproduction de la Cigogne blanche (*Ciconia ciconia*, Aves, Ciconiidae) et du Héron garde-bœufs (*Ardea ibis*, Aves, Ardeidae) dans la région de Tébessa. <http://docplayer.fr/72614526-Doctorat-en-sciences.html>
- [18]. wilaya de Tebessa. (s. d.). drc batna. Consulté le 9 mars 2022, à l'adresse <https://drcbatna.dz/index.php/homepage?id=75:2013-06-18-10-17-43&catid=78>
- [19]. ZOUINI Derradji. (2011). GESTION INTÉGRÉE DES EAUX DU BASSIN D'EL MALABIOD ABIOD W., TÉBESSA, NORD ORD-EST ALGÉRIENE. ;UNIVERSITÉ BADJI MOKHTAR-ANNABA. <https://docplayer.fr/141290505-Abiod-w-de-tebessa-nord.html>
- [20]. UTILISATION DE L'APPRENTISSAGE PROFOND POUR LA CLASSIFICATION DES MAUVAISES HERBES DANS LE BLEUET NAIN. (2021, avril). Université du Québec à Trois-Rivières. <https://depot-e.uqtr.ca/id/eprint/9839/1/eprint9839.pdf>
- [21]. 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
- [22]. Infos, A. (2020, août 12). agriculture-satellite. Algérie infos. <https://www.algerieinfos.com/agriculture-lalgerie-mise-sur-la-technologie-geospatiale-pour-ses-grands-projets-dans-le-sud/agriculture-satellite/>
- [23]. R\U00E9Daction, L. (s. d.). Intelligence artificielle : définition, conseils, comparatifs, témoignages. . . intelligence artificiel. [https://www.journaldunet.com/solutions/dsi/1493139-intelligence-artificielle/#:%7E:text=L%27intelligence%20artificielle%20\(IA\),de%20la%20m%C3%A9moire%20et%20l%27](https://www.journaldunet.com/solutions/dsi/1493139-intelligence-artificielle/#:%7E:text=L%27intelligence%20artificielle%20(IA),de%20la%20m%C3%A9moire%20et%20l%27)
- [24]. La Classification d'images d'insectes ravageurs en utilisant le Deep Learning. (2019–2020). <https://www.ummo.dz/dspace/bitstream/handle/ummo/13208/Lounis%20K.%3B%20Moussi%20D..pdf?sequence=1&isAllowed=y>

- [25]. Claude Touzet. LES RESEAUX DE NEURONES ARTIFICIELS, INTRODUCTION AU CONNEXIONNISME : COURS, EXERCICES ET TRAVAUX PRATIQUES. EC2, 1992, Collection de l'EERIE,N. Giambiasi. fhal-01338010f
- [26]. Démystifier le Machine Learning, Partie 2 : les Réseaux de Neurones artificiels. (s. d.). Juri'Predis. <https://www.juripredis.com/fr/blog/id-19-demystifier-le-machine-learning-partie-2-les-reseaux-de-neurones-artificiels>
- [27]. SPRH LABS. (2021, 9 décembre). Understanding Deep Learning : DNN, RNN, LSTM, CNN and R-CNN. Medium. <https://medium.com/@sprhlibs/understanding-deep-learning-dnn-rnn-lstm-cnn-and-r-cnn-6602ed94dbff>
- [28]. Karpathy, A. (2017). Convolutional Neural Networks (CNNs / ConvNets). 10 juillet 2018, `à partir de <http://cs231n.github.io/convolutional-networks/>
- [29]. Lee, J., Kim, T., Park, J. & Nam, J. (2017). Raw Waveform-based Audio Classification Using Sample-level CNN Architectures. Corr., abs/1712.00866. arXiv1712.00866. Récupérée `à partir de <http://arxiv.org/abs/1712.00866>
- [30]. Lopez, M. M. & Kalita, J. (2017). Deep Learning applied to NLP. CoRR, abs/1703.03091.arXiv : 1703.03091. Récupérée `à partir de <http://arxiv.org/abs/1703.03091>
- [31]. Touzet, C. (1992). Les réseaux de neurones artificiels, introduction au connexionnisme.EC2.
- [32]. Controledesmauvaisesherbes. (2017, 25 janvier). Controle des mauvaises herbes. Consulté le 2022, à l'adresse https://www.mapaq.gouv.qc.ca/SiteCollectionDocuments/Regions/LavalLanaudiere/Journee_sagricoles2017/Controledesmauvaisesherbes.pdf
- [33]. Dr Merzougui. (201apr. J.-C.). Deep learning. http://staff.univ-batna2.dz/sites/default/files/merzougui_ghalia/files/support_de_cours_-deep_learning-chapitre3-cnn.pdf
- [34]. Girshick, R. B., Donahue, J., Darrell, T. & Malik, J. (2013). Rich feature hierarchies for accurate object detection and semantic segmentation. CoRR, abs/1311.2524.arXiv : 1311.2524. Récupérée `a partir de <http://arxiv.org/abs/1311.2524>
- [35]. Vincent Schülé Sous La Direction De Prof. Dr Dominique Genoud. (2018, août 24). Détection automatique d'objets agricoles avec du Machine Learning. object detection. Consulté le 2022, à l'adresse https://www.hevs.ch/media/document/3/schule_vincent_master.pdf
- [36]. CS 230 - Pense-bÃate de rÃ©seaux de neurones convolutionnels. (s. d.). R-cNN. Consulté le 2022, à l'adresse <https://stanford.edu/%7Eshervine/l/fr/teaching/cs-230/pense-bete-reseaux-neurones-convolutionnels>
- [37]. Introduction to YOLO Algorithm for Object Detection. (s. d.). Engineering Education (EngEd) Program | Section. Consulté le 2022, à l'adresse

<https://www.section.io/engineering-education/introduction-to-yolo-algorithm-for-object-detection/#:%7E:text=YOLO%20algorithm%20employs%20convolutional%20neural,in%20a%20single%20algorithm%20run>

- [38]. Ren, S., He, K., Girshick, R. B. & Sun, J. (2015). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. CoRR, abs/1506.01497. Ari : 1506.01497. R´ecup´er´ee `à partir de <http://arxiv.org/abs/1506.01497>
- [39]. Dai, J., He, K. & Sun, J. (2015). Instance-aware Semantic Segmentation via Multi-task Network Cascades. CoRR, abs/1512.04412. arXiv : 1512.04412. R´ecup´er´ee `à partir de <http://arxiv.org/abs/1512.04412>
- [40]. JOHNSON, Femi Temitope et AKERELE, Joel. Artificial Intelligence For Weed Detection-A Techno-Efficient Approach. ICTACT Journal on Image and Video Processing, 2020, vol. 11, no 2, p. 2299-2305.
- [41]. A. Latha, B.V. Poojith and G. Vittal Kumar, “Image Processing in Agriculture”, International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering, Vol. 2, No. 6, pp.1562-1565, 2014.
- [42]. Michelle Araujo E Viegas, Avinash Kurian, Victor Joshua Rebello and Niraj Mangaldas Gaunker, “Weed Detection using Image Processing”, International Journal for Scientific Research and Development, Vol. 4, No. 11, pp. 660-662, 2017.
- [43]. Shraddha S. Durugkar, Priyanka S. Jadhav, Shraddha S.Zade and Vijay S. Bhong, “Design of Farming Robot for Weed Detection and Herbicides Applications Using Image ISSN: 0976-9102 (ONLINE) ICTACT JOURNAL ON IMAGE AND VIDEO PROCESSING, NOVEMBER 2020, VOLUME: 11, ISSUE: 022305 Processing”, International Journal of Research in Engineering and Technology, Vol. 4, No. 3, pp. 161-163, 2018.
- [44]. D.S. Radhika Shetty and G.K. Roopa, “Weed Detection using Image Filtering in Vegetables Crops”, IOSR Journal of Computer Engineering, Vol. 21, No. 1, pp. 61-64, 2019
- [45]. K. Sandeep Kumar, Rajeswari and B.N. Usha, “Convolution Neural Network Based Weed Detection in Horticulture Plantation”, International Journal of Scientific Research and Review, Vol. 7, No. 6, pp. 41-47, 2018.
- [46]. Adel Bakhshipoura and Abdolabbas Jafarib, “Evaluation of Support Vector Machine and Artificial Neural Networks in Weed Detection using Shape Features”, Computers and Electronics in Agriculture, Vol. 145, pp.153-160, 2018.
- [47]. G.L. Grinblat, L.C. Uzal, M.G. Larese and P.M. Granitto, “Deep Learning for Plant Identification using Vein Morphological Patterns”, Computers and Electronics in Agriculture, Vol. 127, pp. 418-424, 2016.

- [48]. Yanlei Xu, Run He, Zongmei Gao, Chenxiao L, Yuting Zhai and Yubin Jiao, “Weed Density Detection Method Based on Absolute Feature Corner Points in Field”, MDPI Journal of Agronomy, Vol. 10, pp. 1-20, 2020.
- [49]. Lin et al., 2017 Lin, Tsung-Yi et al. Feature pyramid networks for object detection. In: Conference on Computer Vision and Pattern Recognition (CVPR). 2017
- [50]. Ricaurte-Quijano, P., & Carli-Álvarez, A. (2016). The Wiki Learning Project : Wikipedia as an Open Learning Environment. *Comunicar*, 24(49), 61-69. <https://doi.org/10.3916/c49-2016-06>
- [51]. Wikipedia contributors. (2020, 27 juillet). Keras. keras. Consulté le 2022, à l’adresse <https://fr.wikipedia.org/wiki/Keras>
- [52]. Wikipedia contributors. (2021, 6 octobre). TensorFlow. tf. Consulté le 2022, à l’adresse <https://fr.wikipedia.org/wiki/TensorFlow>
- [53]. Wikipedia contributors. (2022, 20 avril). NumPy. np. Consulté le 2022, à l’adresse <https://fr.wikipedia.org/wiki/NumPy>
- [54]. Wikipedia contributors. (2022a, avril 20). Matplotlib. m. Consulté le 2022, à l’adresse <https://fr.wikipedia.org/wiki/Matplotlib>
- [55]. Wikipedia contributors. (2020b, août 27). Scikit-image. img. Consulté le 2022, à l’adresse <https://fr.wikipedia.org/wiki/Scikit-image>
- [56]. Wikipedia contributors. (2021b, novembre 8). OpenCV. cv. Consulté le 2022, à l’adresse <https://fr.wikipedia.org/wiki/OpenCV>
- [57]. Visual Geometry Group - University of Oxford. (s. d.). VGG. Consulté le 2022, à l’adresse <https://www.robots.ox.ac.uk/%7Evvgg/software/via/>
- [58]. Computer Vision Annotation Formats. (s. d.). ROBOFLOW. Consulté le 2022, à l’adresse <https://roboflow.com/formats>
- [59]. Wikipedia contributors. (2021b, octobre 6). TensorFlow. tf. Consulté le 2022, à l’adresse <https://fr.wikipedia.org/wiki/TensorFlow>
- [60]. Google Colaboratory. (s. d.). Colab. Consulté le 2022, à l’adresse <https://colab.research.google.com/>
- [61]. IA. (2022). [Graphique]. <https://www.cardio-online.fr/var/cardio/storage/images/media/cardio/images/dossier-speciaux/intelligence-artificielle-et-echocardiographie/diagramme-de-venn/6223627-1-fre-FR/Diagramme-de-Venn.jpg>