

الجمهورية الجزائرية الديمقراطية الشعبية

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



UNIVERSITE MOHAMED SEDDIK BENYAHIA JIJEL

Faculté des Sciences et de la Technologie

Département d'Electronique

N° :/2022

MEMOIRE DE MASTER

DOMAINE : Sciences et Technologies

FILIERE : Electronique

SPECIALITE : Electronique des Systèmes Embarqués

Thème

Conception et réalisation d'un système semi-automatique de diagnostic des générateurs photovoltaïques à l'aide de l'apprentissage profond et de l'imagerie thermique

Présenté Par : Chourouk MOUSSAOUI

Encadré Par : Pr. Adel MELLIT

Date de soutenance : 12 /07/2022

Jury de Soutenance

Président : A. SOUKKOU

Grade MCA

Univ. MSB Jijel

Encadreur : A. MELLIT

Grade Pr

Univ. MSB Jijel

Co-Encadreur : N. KELLIL

Grade AR

UDES, Tipaza

Examineur : M. BOUZERDOUM

Grade MCB

Univ. MSB Jijel

Promotion : 2021 /2022

في هذه المذكرة ، يتم تقديم نظام شبه آلي لاكتشاف الأخطاء وتصنيف المصفوفات الكهروضوئية (PV) بناءً على صور الأشعة تحت الحمراء (التصوير الحراري) وشبكة عصبية عميقة (DCNN). تم تطوير مصنف ثنائي يعتمد على DCNN للكشف عن الخطأ، ثم تم تطوير مصنف آخر متعدد الطبقات يعتمد على DCNN لتحديد نوع الخطأ على مصفوفة PV. الأعطال التي تم فحصها هي تأثير التظليل الجزئي، وتراكم الغبار على سطح الوحدة الكهروضوئية، ووحدة الكهروضوئية مستقصرة الدارة، والصمام الثنائي المفتوح في وحدة الكهروضوئية. ثم تم تضمين المصنفات المطورة القائمة على DCNN في معالج دقيق (Raspberry Pi 4). يتم إخطار المستخدم بحالة نظام PV بواسطة الرسائل القصيرة والبريد الإلكتروني. أظهرت المحاكاة والنتائج التجريبية قدرة المصنفات المطورة على اكتشاف وتصنيف الخطأ بدقة جيدة جدًا تبلغ 92% و98% للكشف والتصنيف على التوالي.

Résumé

Dans ce mémoire, un système semi-automatique de détection et classification des défauts dans un générateur photovoltaïques (PV) basé sur des images infrarouges et un réseau neuronal conventionnel profond (DCNN) est présenté. Deux classificateurs ont été développés, un classificateur binaire pour détecter le défaut, et un autre classificateur multi-classe est également développé pour la classification du type de défaut qui s'est survenu sur le générateur PV. Les défauts étudiés sont l'effet d'ombrage partiel, l'accumulation de poussière sur la surface du module PV, le module PV court-circuité et la diode bypass dans un module PV. Les classificateurs développés ont été intégrés dans un microprocesseur (Raspberry Pi 4). L'utilisateur est informé de l'état du système PV par un SMS et un e-mail. Les résultats de simulation et expérimentaux ont montré que les classificateurs développés étaient capables de détecter et de classer les défauts avec une précision de 98 % pour la détection et 92% pour la classification,

Mots clés : Photovoltaïque, CNN, DCNN, classification, apprentissage profond, Raspberry Pi.

Abstract

In this thesis, a semi-automatic system for fault detection and classification of photovoltaic (PV) arrays based on infrared (thermography) images and a deep conventional neural network (DCNN) is presented. A binary classifier based on a DCNN is developed to detect the fault, then another multiclass classifier based on a DCNN is also developed for the identification of the type of fault that occurred on a PV array. The investigated faults are partial shading effect, dust accumulation on PV modules surface, short-circuited PV module, and open-circuited diode in a PV module. The developed DCNN-based classifiers have been then embedded into a microprocessor (Raspberry Pi 4). The user is notified about the state of the PV system by SMS and email. Simulation and experimental results showed the ability of the developed classifiers to detect and classify fault with a quite good accuracy of 98% and 92 % for the detection and classification, respectively.

Keywords: Photovoltaic, CNN, DCNN, classification, deep learning, Raspberry Pi.

Remerciements



Au terme de ce travail, je voudrais tout d'abord remercier **ALLAH** de m'avoir donné la santé, la volonté, la patience et le courage pour réaliser ce projet.

J'exprime toute ma reconnaissance à mon encadreur, Monsieur A. MELLIT, pour son aide précieuse sans laquelle ce travail n'aurait pas vu le jour.

Mes remerciements s'adressent à M. N. KELLIL pour son aide à la préparation de la base de données que nous avons utilisé dans ce travail.

De plus, je tiens à remercier les membres de jury Dr. A. SOUKKOU et Dr. M. BOUZERDOUME pour accepter d'évaluer ce travail.

Je voudrais aussi remercier toutes les personnes qui ont participé de près ou de loin à l'élaboration de ce mémoire, notamment :

Mme. S. BOUATMANE, pour sa disponibilité, son aide et ses conseils.

M. O. HERRAK (ex-Master de M. MELLIT) et M. LAISSAOUI , pour leurs conseils et leur aide dans la partie pratique.

Enfin, le plus grand remerciement s'adresse à moi-même Chourouk MOUSSAOUI pour ma patience, volonté et mes efforts tout au long la réalisation de ce projet.

Moussaoui Shourouk

Dédicace



Je dédie ce modeste travail à ceux qui, quels que soient les termes embrassés, je n'arriverais jamais à leur exprimer mon amour sincère.

A mon exemple, mon précieux offre du dieu, qui doit ma vie, qui n'a jamais cessé de formuler des pierres à mon égard, de me soutenir et de m'épauler pour que je puisse atteindre mes objectifs ma réussite, tout mon respect : mon cher père S. MOUSSAOUI.

A la femme qui a souffert sans me laisser souffrir, qui n'a jamais dit non à mes exigences et qui n'a épargné aucun effort pour me rendre heureuse : ma chère mère F. LAIB.

A ma chère sœur Nada et mes frères Fdjrou, Iyad et Taqui pour leur existence dans ma vie et pour leur encouragement, que Dieu leur donne une joyeuse vie pleine du succès.

A ma chère tante K. LAIB, son mari R. MADJ et leur fils Mohammed pour leur soutien, encouragement et support dans les moments difficiles, que Dieu les protège et leur donne une longue et joyeuse vie.

A mes amies qui n'ont pas ménagé leurs efforts et qui m'ont soutenu tout au long de ce travail, particulièrement : Y. AMIAR, A. KIHHEL, NH. AZZOUZA, R. BOUCHEROUT, S. BOUSSUAR, S. HADDAD, K. KERMIMOUN, Z. NOUALI, N. AZZOUZI, IS. MADOU, C. HEDJAB, Z. REDJEL, W. BOUZIANE, N. YAAKOUBI, I. ABED, D. JBABRA, A. BAOUNI, S. BEN YAMINA et I, MOUSSAOUI que Dieu les protège et leurs offre le succès et le bonheur.

Table de matières

Résumé.....	i
Remerciement.....	ii
Dédicace	iii
Table de matières.....	iv
Liste des abréviations.....	viii
Listes des figures.....	ix
Liste des tableaux.....	xi
Introduction générale.....	1

Chapitre 1

Diagnostic de défauts des générateurs photovoltaïques.

1.1. Introduction.....	3
1.2. L'effet photovoltaïque.....	3
1.3. Généralités.....	3
1.3.1. Cellule PV.....	3
1.3.2. Le module photovoltaïque.....	4
1.3.3. Générateur photovoltaïque.....	5
1.3.4. Chaîne photovoltaïque.....	5
1.3.5. Champ photovoltaïque.....	5
1.3.6. Protection des modules photovoltaïques.....	5
1.4. Les défauts des modules photovoltaïques.....	7
1.4.1. Les bulles d'air.....	7
1.4.2. Fissure dans les cellules.....	8
1.4.3. Délaminage.....	8
1.4.4. Points Chauds (Hot spots)	9
1.4.5. Défaut de mis match et d'ombrage partiel.....	10
1.4.6. Défauts de la diode de by-pass et diode anti-retour.....	11

1.5.	Méthodes de diagnostic des défauts.....	12
	1.5.1. Méthodes électriques.....	12
	1.5.2. Méthodes non-électriques.....	13
1.6.	Conclusion.....	14

Chapitre 2

Méthodes de classification d'images.

2.1.	Introduction.....	15
2.2.	Traitement d'image.....	15
	2.2.1. Définitions.....	15
	2.2.1.1. L'image.....	15
	2.2.1.2. Traitement d'image.....	15
	2.2.1.3. L'histogramme.....	16
	2.2.1.4. Quelques techniques de traitement d'image.....	16
2.3.	Classification des images.....	18
	2.3.1. Apprentissage automatique (Machine Learning).....	19
	2.3.2. Apprentissage profond (Deep learning).....	21
	2.3.2.1.Types d'apprentissage.....	22
	2.3.2.2.Réseaux de neurone convolutifs profonds.....	24
	i. Historique.....	24
	ii. Architecture de base.....	26
	iii. Les couches principales d'un DCNN.....	27
	iv. Exemples des DCNN les plus utilisés.....	29
2.4.	Conclusion.....	31

Chapitre 3

Mise en œuvre d'une méthode de diagnostic des générateurs photovoltaïques.

3.1.	Introduction.....	32
------	-------------------	----

3.2.	Ressources.....	32
3.2.1.	Ressources matérielles.....	32
3.2.2.	Ressources logicielles.....	32
3.3.	Application de l'apprentissage profond (DL) pour le diagnostic des générateurs PVs.....	35
3.4.	Méthodologie.....	36
3.4.1.	Description.....	36
3.4.2.	Base de données utilisée.....	36
3.4.3.	Développement du deux modèles.....	38
3.4.4.	Architecture.....	39
3.4.5.	Développement de l'interface graphique.....	41
3.5.	Conclusion.....	42

Chapitre 4

Test et résultats expérimentaux.

4.1.	Introduction.....	43
4.2.	Méthodes d'évaluation.....	43
4.2.1.	Matrice de confusion.....	43
4.2.2.	Courbe ROC et AUC.....	44
4.3.	Matériel utilisé.....	44
4.4.	Résultats d'apprentissage et validation.....	44
4.4.1.	La détection.....	44
4.4.1.1.	Précision d'apprentissage et validation.....	44
4.4.1.2.	Matrice de confusion.....	46
4.4.1.3.	Courbe ROC.....	46
4.4.2.	La classification.....	47
4.4.2.1.	Précision d'apprentissage et validation.....	47
4.4.2.2.	Matrice de confusion.....	48
4.4.2.3.	Courbe ROC.....	49
4.5.	Résultats du test et simulation.....	50

4.5.1. Sauvegarde et optimisation des modèles.....	50
4.5.2. Test du programme principal.....	50
4.5.3. Stockage des données.....	51
4.6. Utilisation de l'interface graphique.....	52
4.7. Comparaison avec d'autres méthodes.....	55
4.8. Conclusion.....	55
Conclusion générale.....	57
Références.....	59

Liste des abréviations

DL	: Deep Learning.
ML	: Machine Learning.
CNN	: Réseaux de neurones convolutifs (Convolutionnal Neural Network).
DCNN	: Réseaux de neurones convolutifs profonds (Deep Convolutionnal Neural Network).
DU	: Dust deposit on PV module.
SC	: Short Circuited PV module.
SD	: Partiel Shading effect.
BD	: Damaged diode bypass.

Liste des figures

Figure 1.1. Structure basique d'une cellule photovoltaïque.....	4
Figure 1.2. Module photovoltaïque.....	5
Figure 1.3. Fonctionnement de la diode bypass.....	6
Figure 1.4. Schéma d'un champ photovoltaïque avec diodes by-pass et diode anti-retour.....	7
Figure 1.5. Défaut de bulle d'air.....	8
Figure 1.6. Les fissures dans les cellules PV.....	8
Figure 1.7. Défaut de délaminage.....	9
Figure 1.8. Défaut de Hot spots.....	10
Figure 1.9. Défaut d'ombrage partiel.....	11
Figure 1.10. Défaut de diode by-pass.....	11
Figure 1.11. Caractéristiques (IV, PV) du module photovoltaïque sous Matlab-Simulink....	13
Figure 2.1. Exemple d'un histogramme d'une image fait par Matlab.....	16
Figure 2.2. Les différents traitements qui peuvent être appliqués sur une image.....	18
Figure 2.3. Les méthodes de classification des images.....	19
Figure 2.4. La famille de l'apprentissage profond.....	22
Figure 2.5. Perceptrons multicouches.....	25
Figure 2.6. Architecture basic.....	26
Figure 2.7. Exemple d'une opération AveragePooling.....	28
Figure 2.8. Exemple d'une opération MaxPooling.....	28
Figure 3.1. Une page de coder sur Colab.....	33
Figure 3.2. Interface Anaconda.....	34
Figure 3.3. Les étapes de travaille.....	36
Figure 3.4. Système PV et caméra FLIR.....	36
Figure 3.5. Base de données pour la détection.....	37
Figure 3.6. Base de données pour la classification.....	37

Figure 3.7. Echantillons aléatoires.....	39
Figure 3.8. Organigramme du fonctionnement.....	41
Figure 4.1. Format de la matrice de confusion.....	43
Figure 4.2. La fonction Coste (loss) et la précision de l'apprentissage pour la détection.....	45
Figure 4.3. La fonction loss et la précision de la validation pour la détection.....	45
Figure 4.4. La matrice de confusion de la détection.....	46
Figure 4.5. La courbe ROC pour la détection.....	47
Figure 4.6. La fonction Coste (loss) et la précision de l'apprentissage pour la classification.	47
Figure 4.7. Les pertes et la précision de la validation pour la classification.....	48
Figure 4.8. La matrice de confusion de la classification.....	49
Figure 4.9. La courbe ROC-AUC de la classification.....	49
Figure 4.10. Réalisation et vérification d'envoi des messages.....	50
Figure 4.11. Firebase.....	51
Figure 4.12. Fonctionnement du système.....	52
Figure 4.13. L'interface Graphique.....	52
Figure 4.14. Affichage sur l'interface.....	54
Figure 4.15. Test d'une nouvelle image.....	55

Liste des tableaux

Tableau 2.1. Bref aperçu des architectures CNN.....	29
Tableau 3.1. Applications de DL dans le diagnostic des générateurs PV du 2018 à 2020.....	35
Tableau 3.2. Architecture et paramètres du modèle de la détection.....	39
Tableau 3.3. Architecture et paramètres du modèle de la classification.....	40
Tableau 4.1. Précision, rappel et f1-score de la détection.....	46
Tableau 4.2. Précision, rappel et f1-score de la classification.....	48
Tableau 4.3. Résultats de la comparaison.....	55



Introduction générale



Introduction générale

Le plus grand problème auquel l'humanité est confrontée à notre époque est le problème de l'énergie durable et parmi les solutions proposées l'énergie solaire photovoltaïque qui représente le future de l'énergie renouvelable.

Un système photovoltaïque (PV) comprend un ou plusieurs modules solaires combinés à un onduleur et à d'autres équipements électriques qui utilisent l'énergie du Soleil pour produire de l'électricité. La taille des systèmes PVs peut varier considérablement, allant de petits systèmes portatifs sur les toits à d'énormes centrales de production à l'échelle des services publics [1].

Il est bien connu que les défauts de tout composant d'une centrale PV réduise considérablement ses performances et donc sa puissance de sortie. Le gros problème est le risque d'incendie qui est principalement dû aux phénomènes de points chauds dans les cellules solaires. À cet égard, diverses méthodes ont été développées pour protéger et isoler les défauts des réseaux PV [2]. La plupart des défauts sont survenues sur le générateur PV, comme l'accumulation de poussière sur la surface PV, le vieillissement de cellules, l'effet d'ombrage partiel et les défauts internes.

De nos jours, de nombreux travaux ont été faites pour localiser, détecter et diagnostiquer les installations PV à partir d'images recueillies (images infrarouge, électroluminescence, ... etc) [3], et récemment, un bon nombre d'articles ont démontré l'utilisation de la technique de l'intelligence artificielle (IA), y compris l'apprentissage profond (DL) pour détecter et diagnostiquer les défauts à partir d'images thermographiques [4-5]. Actuellement, l'application de réseaux neuronaux conventionnels profonds (DCNNs) est la technique la plus utilisée dans ce domaine pour résoudre ce problème [6].

La contribution principale de ce mémoire est de concevoir un système semi-automatique de diagnostic des générateurs/modules PV en utilisant un DCNN et des images thermographiques. Les algorithmes de diagnostic seront optimisés et intégrés dans un microprocesseur (Raspberry Pi 4) pour une application en temps réel. Le système semi-automatique proposé peut détecter, classer et notifier les utilisateurs concernant le type de défaut qui s'est produit sur leurs générateur PV. Ce travail a été réalisé au niveau du laboratoire des énergies renouvelables, université de jijel.

Il est à noter que la base de données (images thermiques) utilisée dans ce travail a été préparée et collectée au niveau du centre de recherche, l'UDES, Bousmail, Tipaza (Camera Professional FLIR T-540). Afin de vérifier la généralisation des classificateurs développés nous allons utiliser quelques images thermiques (PV modules disponible dans le Hall de Technologie, Université de Jijel) capturées par une caméra thermique (FLIR-ONE, pour application Téléphone).

Le mémoire est organisé comme suit :

Le chapitre 1 est consacré au diagnostic des défauts des générateurs photovoltaïques. En introduisant des généralités sur ces générateurs PVs, et présentant les défauts et quelques méthodes de diagnostic.

Le chapitre 2 est consacré aux méthodes de classification, en abordant les types d'apprentissage et une brève introduction aux réseaux de neurone convolutifs.

Dans le chapitre 3 nous allons mettre en œuvre notre travail en énumérant les étapes que nous avons suivies.

Nous présenterons les résultats obtenus dans le chapitre 4, ces résultats sont : ceux de la phase d'apprentissage, du test et d'affichage sur l'interface graphique développée.



Chapitre 1

Diagnostic de défauts des générateurs photovoltaïques



1.1. Introduction

Dans ce chapitre nous allons entamer les points suivants :

- L'effet photovoltaïque.
- Généralités.
- Les défauts des modules photovoltaïques.
- Méthodes de diagnostic des défauts.

1.2. L'effet photovoltaïque

L'effet photovoltaïque, découvert en 1839 par le français Edmond Becquerel, désigne la capacité que possèdent certains matériaux, notamment les semi-conducteurs, à convertir directement les différentes composantes de la lumière du soleil (et non sa chaleur) en électricité. [7].

1.3. Généralités

1.3.1. Cellule PV

Une cellule photovoltaïque (PV) est une technique de collecte d'énergie qui transforme l'énergie solaire en électricité utilisable grâce à un phénomène appelé "effet photovoltaïque". Il existe plusieurs types de cellules photovoltaïques qui utilisent toutes des semi-conducteurs pour interagir avec les photons du soleil afin de générer un courant électrique [8].

La cellule photovoltaïque est composée de plusieurs couches de matériaux, chacune ayant une fonction spécifique. La couche la plus importante d'une cellule photovoltaïque est la couche semi-conductrice spécialement traitée. Elle est composée de deux couches distinctes (type p et type n - voir figure 1.1), et c'est elle qui convertit réellement l'énergie du soleil en électricité utile par un processus appelé effet photovoltaïque. De chaque côté du semi-conducteur se trouve une couche de matériau conducteur qui "collecte" l'électricité produite. Notez que la face arrière ou ombragée de la cellule peut se permettre d'être entièrement recouverte par le conducteur, tandis que la face avant ou éclairée doit utiliser les conducteurs avec parcimonie pour éviter de bloquer une trop grande partie du rayonnement solaire pour qu'il n'atteigne le semi-conducteur. La dernière couche qui est appliquée uniquement sur le côté éclairé de la cellule est le revêtement antireflet. Comme tous les semi-conducteurs sont naturellement réfléchissants, la perte par réflexion peut être

importante. La solution consiste à utiliser une ou plusieurs couches d'un revêtement anti-reflet (similaire à ceux utilisés pour les lunettes et les appareils photo) pour réduire la quantité de rayonnement solaire qui est réfléchi à la surface de la cellule [8].

Ils existent plusieurs types de cellules PV tels que :

- La cellule monocristalline.
- La cellule polycristallin.
- La cellule amorphe à couche mince.
- La cellule solaire HIT.

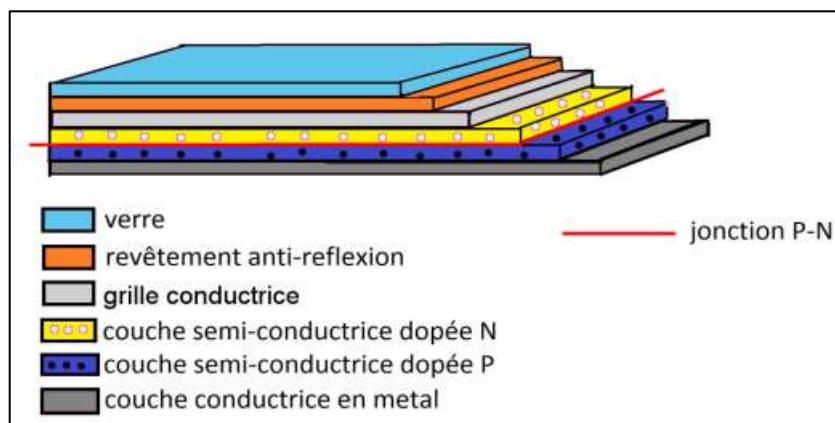


Figure 1.1. Structure basique d'une cellule photovoltaïque [8].

1.3.2. Le module photovoltaïque

L'énergie produite par une cellule photovoltaïque est insuffisante pour un usage général. Normalement une seule cellule photovoltaïque produit une tension et une puissance de sortie relativement faibles. Le module PV est constitué par la connexion de groupes de cellules PV en série et en parallèle avec leurs dispositifs protecteurs, qui représentent l'unité de la conversion dans le système de production. Par la connexion en série de nombreuses cellules ; la tension du module PV est augmentée et les pertes de conduction dans les câbles sont réduites au minimum, alors que la connexion en parallèle des cellules augmente le courant. Un module photovoltaïque est le composant de base de la production d'énergie PV à grande échelle. La majorité des modules commerciaux sont constitués de 36 à 72 cellules. Les modules sont produits d'un panneau transparent, des cellules photovoltaïques encapsulées et d'une face arrière. Le matériau sur la façade est généralement du verre trempé et à faible teneur en fer (figure 1.2). Le rendement d'un module PV est

inférieur à celui d'une cellule PV car une partie de l'irradiation solaire est réfléchi par la couverture en verre et l'ombre portée de cadre [8].



Figure 1.2. Module photovoltaïque [8].

1.3.3. Générateur photovoltaïque

Le générateur photovoltaïque est l'unité de production d'énergie électrique sous forme de courant continu. Le composant élémentaire de cette unité qui convertit l'énergie solaire en l'énergie électrique est la cellule photovoltaïque [9].

1.3.4. Chaîne photovoltaïque

Un string PV, appelé également chaîne PV, est un ensemble de modules connectés en série afin de générer la tension de sortie spécifiée [9].

1.3.5. Champ photovoltaïque

Pour obtenir des puissances de quelques KW, sous une tension adéquate, il faut regrouper les modules en série et en parallèle. Ce regroupement constitue un champ photovoltaïque. Il y'a plusieurs configurations possibles pour l'interconnexion de modules au sein d'un champ photovoltaïque : connexion série parallèle simple, connexion Total Cross Tied, connexion Bridge Linked. [9].

1.3.6. Protection des modules photovoltaïques

Comme tout système électrique, les systèmes photovoltaïques doivent être protégés contre certains problèmes qui peuvent obstruer le fonctionnement de ces systèmes ; pour éviter ces problèmes ils existent plusieurs moyens.

- **Diode bypass**

Comme c'est montré dans la figure 1.3, la cellule ombrée cesse la production de l'énergie électrique et se comporte comme une résistance semi-conductrice. Donc elle génère

moins de courant que les deux autres cellules, qui la forte diminution de la production d'énergie de la chaîne. Donc la puissance générée par les cellules non-ombrées (ensoleillées) sera dissipée par la cellule ombrée, cette cellule peut avec le temps provoquer un point chaud et éventuellement la destruction de la mauvaise cellule [8].

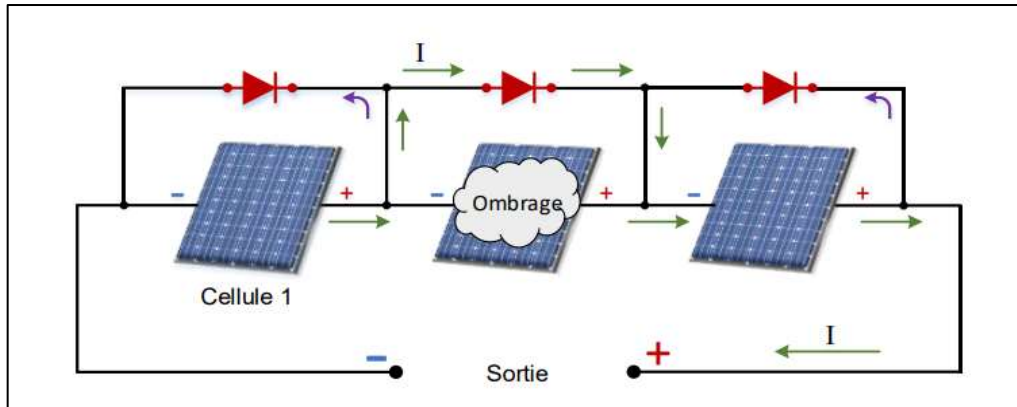


Figure 1.3. Fonctionnement de la diode bypass [8].

Les diodes de by-pass sont reliées à l'extérieur et en antiparallèle avec chaque cellule PV pour apporter un chemin électrique alternatif pour le courant généré. Parce qu'il ne peut pas traverser la cellule ombrée. Cela aide à préserver les performances de la chaîne en limitant la tension de polarisation inverse générée à travers n'importe quelle cellule partiellement ombrée et donc de réduire la puissance électrique qui peut être dissipée par la cellule [8].

- **Diode anti-retour**

La tension construite par chaque string peut être différente. Pendant la mise en parallèle de ces strings pour former un champ, le string avec la tension la plus faible peut absorber un courant inverse provenant des autres strings. Cela conduit donc à une baisse de production et les modules du string traversés par le courant inverse pourraient être également susceptibles de la défaillance. Pour éviter ces courants inverses, une diode anti-retour est placée au bout de chaque string [9].

Cependant, l'utilisation de la diode anti-retour entraîne une perte dans la production du fait de la chute de tension provoquée par cette diode lors du fonctionnement normal du champ photovoltaïque. En outre, ces diodes peuvent entrer en défaut et ont donc besoin d'un contrôle régulier. On utilise parfois un fusible plutôt que la diode anti-retour. Cependant, l'utilisation du fusible ne protège pas le string contre le courant inverse. Le fusible doit être dimensionné de sorte que les composants du string peuvent prendre en charge le

courant si celui-ci existe. Lors l'utilisation du fusible le choix de l'un des deux composants réside donc sur la tolérance entre la perte en fonctionnement normal (pour le cas d'utilisation de la diode anti-retour) et la perte causée par le courant inverse [9].

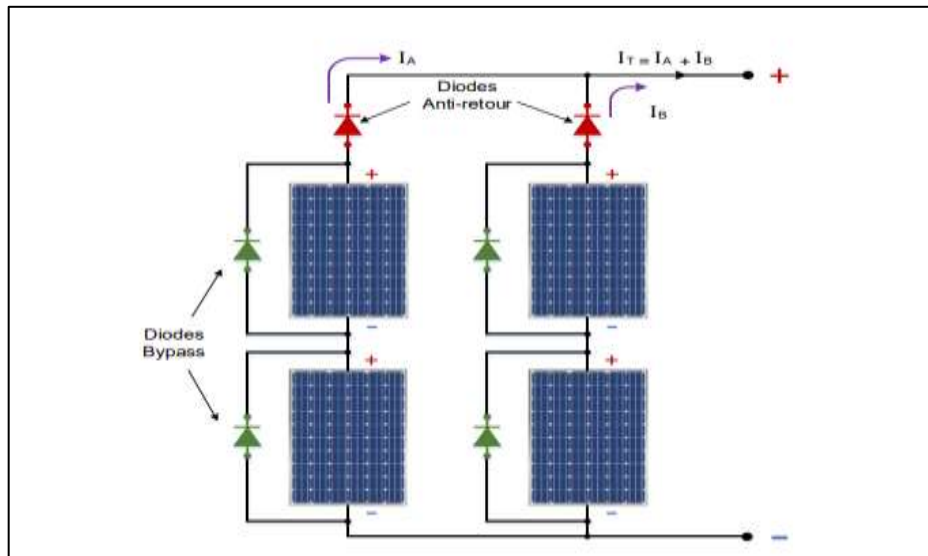


Figure 1.4. Schéma d'un champ photovoltaïque avec diodes by-pass et diode anti-retour [9].

1.4. Les défauts des modules photovoltaïques

Dans la partie précédente nous avons présenté brièvement les systèmes photovoltaïques et leurs principales unités. Durant son fonctionnement, une installation PV peut être soumise à des défauts et conditions qui peuvent provoquer l'endommagement de cette installation, dans cette partie-là nous allons présenter les différents types de ces défauts.

1.4.1. Les bulles d'air

Les bulles sont principalement dues aux réactions chimiques qui dégagent des gaz emprisonnés dans le module photovoltaïque. Ils forment une chambre d'air où la température du gaz est moins élevée que celle dans les cellules voisines. Cependant, la chambre d'air aggrave la chaleur, la capacité de dissipation de la cellule voisine de sorte qu'elle surchauffe et présente donc une température plus élevée que dans les cellules adjacentes. En outre, lorsque les bulles apparaissent sur la face avant provoqués par les microfissures sur la surface de la cellule solaire, une réduction de la radiation atteignant la cellule PV se produit créant ainsi un découplage de la lumière et une réflexion accrue. De plus, les bulles peuvent se casser et peuvent endommager la surface d'étanchéité arrière qui cause une infiltration d'humidité. Un seul module peut contenir plusieurs bulles dans la face arrière.

Elles apparaissent habituellement au milieu de la cellule, et cela peut arriver due à un décollement causé par une haute température [10].



Figure 1.5. Défaut de bulle d'air [11].

1.4.2. Fissure dans les cellules

Une fissure de cellule se produit quand les cellules solaires subissent une force ou une contrainte extérieure thermique. De nos jours, de nombreux constructeurs des PV essaient de minimiser l'épaisseur des cellules solaires pour économiser le silicium et réduire les coûts, l'épaisseur a baissé de 300 μm à moins de 200 μm , ou parfois à moins de 100 μm . En plus la réduction de l'épaisseur de la cellule, la surface de la cellule solaire a augmenté à 210 mm x 210 mm. Cette diminution de l'épaisseur et l'augmentation de la surface ont rendu les cellules plus fragiles et sensibles aux fractures durant leur manipulation et leur stockage [10].

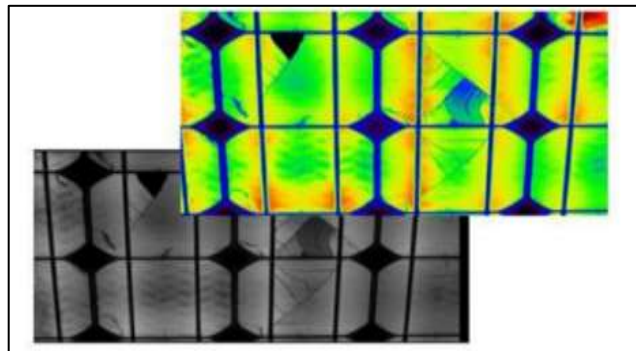


Figure 1.6. Les fissures dans les cellules PV [10].

1.4.3. Délaminage

Au sein d'un panneau, des cellules solaires sont encapsulées avec un matériau d'encapsulation appelé EVA (éthylène acétate de vinyle). L'EVA protège les cellules sensibles de l'influence du climat comme l'humidité et les radiations UV. Lors de la production, les cellules sont laminées d'une manière permanente dans une machine sous vide avec une source de chaleur. Il est important de parvenir la température de polymérisation pour

obtenir un degré élevé possible de réticulation de l'EVA (plus de 70%). Dans le cas du délaminage aux extrémités du module, l'humidité peut s'infiltrer à cause de la pénétration de l'eau à l'intérieur et endommager les cellules et donc provoque la corrosion de la cellule solaire. Aussi bien que la diminution de la lumière, où la réflexion augmente et enfin une perte de la puissance. Le délaminage est plus courant sous les climats chauds et humides. Le résultat s'en trouve souvent la corrosion des matériaux qui constituent la structure du module. Dans ce cas, la résistance électrique PV pourrait augmenter ou une panne de courant pourrait apparaître [10].



Figure 1.7. Défaut de délaminage [11].

1.4.4. Points chauds (Hot spots)

Des cellules ombragées ou fissurées peuvent provoquer un phénomène qui surchauffe localement le module causant sa perte. Ce phénomène ne s'appelle point chaud. Ce phénomène des points chauds est un problème relativement fréquent dans les générateurs photovoltaïques courants. Il comporte à la fois un risque pour la durée de vie du module photovoltaïque et une diminution de son efficacité opérationnelle lorsque des cellules photovoltaïques sont associées en série, le courant circulant dans l'ensemble est limité par le courant généré par les cellules les moins performantes ou les moins éclairées. Ainsi, une cellule qui ne recevrait pas la même quantité de lumière que le reste de l'ensemble, produirait un courant plus faible appelé l'ombrage partiel. Si l'une d'elles était partiellement ombragée le courant ne pouvait plus circuler. La cellule accumulait alors le courant produit en amont et se comportait comme un récepteur. Elle dissipait une puissance sous forme de chaleur pouvant dépasser les 100°C ce qui pouvait la faire griller si le point chaud est causé par un shunt (court-circuit d'une cellule) ou une erreur dans l'interconnexion entre les cellules générant également un shunt. Le point chaud est directement lié à une erreur de fabrication de la cellule ou du module PV. De plus, si une cellule génère

un courant plus faible que le reste. Un point chaud apparaît sous la forme d'une cellule entière fonctionnant à une température plus élevée que le reste des cellules « figure 1.8 » elle peut être considérée aussi comme une erreur de fabrication du module photovoltaïque. Le point chaud pourrait causer des dommages à la cellule ou l'encapsulant dans un court laps de temps de fonctionnement [10].

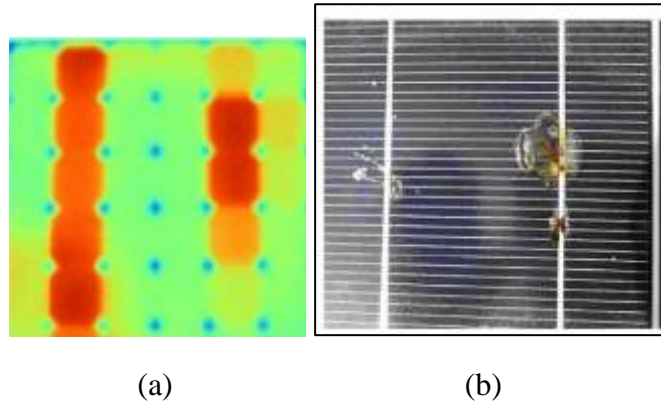


Figure 1.8. Défaut de Hot spots [11].

1.4.5. Défaut de mis match et d'ombrage partiel

Les défauts de mis match sont des défauts fréquents au PV.

- **Mis match**

Le défaut de ‘mis match’ est le défaut provoqué par le groupement de cellules possédant une caractéristique I-V non identique. Tout changement dans l'un des paramètres conduira à la dissemblance de leur caractéristique. Le défaut d'ombrage est un cas particulier du défaut de mis match car sa présence conduit à une réduction de l'ensoleillement reçu par des cellules. Le changement de ces paramètres provient de deux facteurs principaux. Tout d'abord, des cellules pourraient posséder des propriétés physiques différentes suite à une tolérance dans la production. Seule la tolérance de la puissance du module est donnée par les producteurs de la cellule ou du module [10].

- **L'ombrage partiel**

Ils existent deux types d'ombrage : l'ombrage total et l'ombrage partiel. L'ombrage total empêche tout le rayonnement d'atteindre les cellules photovoltaïques, par contre l'ombrage partiel empêche seulement le rayonnement d'atteindre une partie de la cellule (une cheminée, un arbre, poussière, neige,). Chaque générateur photovoltaïque présente un point de travail unique au niveau duquel il peut produire la puissance électrique maximale

possible, appelé le point de puissance maximal (MPP, Maximum Power Point). Les cellules recevant une énergie faible doivent dissiper le surplus de courant délivré par celles qui sont les plus éclairées, créant également des points chauds si le module n'est pas protégé de manière adéquate. Pour éviter la surchauffe de la cellule et les points chauds, et aussi des détériorations définitives, des diodes de protection sont placées sur les générateurs PV de forte puissance (diode by-pass). Cependant, ces protections sont coûteuses en perte du rendement [10].



Figure 1.9. Défaut d'ombrage partiel [10]

1.4.6. Défauts de la diode de by-pass et diode anti-retour

Comme il a été mentionné précédemment la diode de by-pass est une des solutions pour protéger les cellules à l'ombrage partiel. Cette diode se place, généralement, en parallèle avec une branche de dix-huit cellules. On peut constater plusieurs défauts de cette dernière, ces défauts sont en général des défauts de la production. Alors, dans un panneau photovoltaïque on peut trouver une diode déconnectée, une diode équivalente à une impédance quelconque, une diode court-circuitée et une diode inversée. Comme la diode de bypass, la diode anti retour est exposée aux mêmes types de défauts envisagés pour la diode anti-retour : court-circuit, impédance quelconque, circuit ouvert et diode inversée [10].

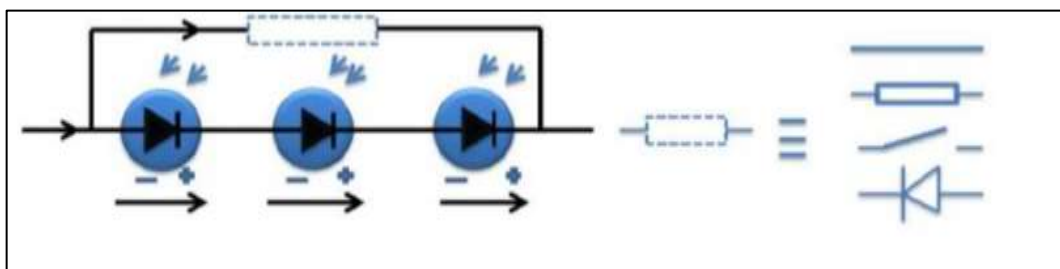


Figure 1.10. Défaut de diode by-pass [10].

1.5. Méthodes de diagnostic des défauts

Ils existent plusieurs méthodes pour diagnostiquer les défauts dans un système PV, soit des méthodes basées sur la mesure et l'analyse des caractéristiques I-V, ce qu'on appelle des méthodes électriques, et des méthodes basées sur l'analyse des grandeurs non-électriques, ce qu'on appelle des méthodes non-électriques.

1.5.1. Méthodes électriques

De nos jours, les systèmes de control d'un système PV, sont essentiellement intégré dans un onduleur, en mesurant les mêmes données (grandeurs) pour tout systèmes PV existants :

- Le courant fournis par le champ PV.
- La tension aux bornes du champ PV.
- La résistance d'isolement entre la borne positive et la borne négative du champ PV.
- En plus, on peut ajouter d'autres paramètres (grandeurs) complémentaires, telles que la température ambiante et l'ensoleillement aux mesures électriques. Pour obtenir les mesures de ces grandeurs on a besoin des capteurs spécifiques [9].

Les mesures AC, sont plus importantes en nombre, car elles ont une relation directe avec l'énergie rendue, ces grandeurs sont :

- Le courant AC.
- La tension AC.
- La fréquence.
- L'impédance du réseau vus par l'onduleur.

D'après toutes ces grandeurs, on peut déduire d'autres :

- La puissance instantanée DC.
- La puissance instantanée AC.
- L'énergies produite [10].

Afin de calculer ces données, un microcontrôleur est intégré dans l'onduleur, ou par d'autres automate qui convient [9].

En termes de diagnostic pour de la maintenance préventive ou curative, il existe très peu d'informations et de fonctionnalités. Au mieux, seule une mesure tension/courant string par string permet d'identifier un éventuel déséquilibre de production et/ou une défaillance d'un onduleur [9].

La figure suivante représente les caractéristiques (I-V, P-V) du module photovoltaïque sous Matlab-Simulink. Les deux paramètres ont été obtenus dans les conditions standard ($G=1000 \text{ W/m}^2$; $T=25^\circ\text{C}$) [12].

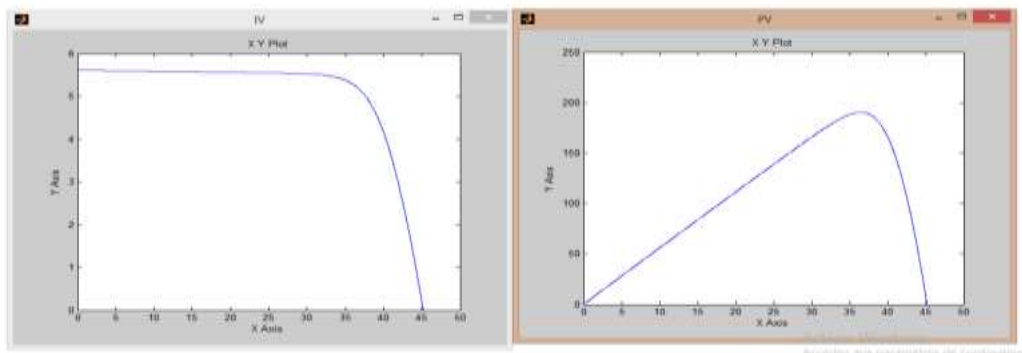


Figure 1.11. Caractéristiques (I-V, P-V) du module photovoltaïque sous Matlab-Simulink [12].

1.5.2. Méthodes non-électriques

Cette méthode est basée soit sur la mécanique, ou le traitement d'image. En intégrant par le traitement d'image, on peut distinguer deux types :

- L'imagerie infrarouge : cette méthode est très répandue pour le diagnostic. Cette méthode est basée sur le concept de génération de chaleur localisée causées par des mauvais contacts, courts circuits [13]. Les paramètres qui doivent être détectés sont la décoloration et le brunissement, les salissures, points chauds, rupture et délaminage des modules PV [8]. On utilise des caméras thermiques pour prendre les images IR afin d'analyser les paramètres cités précédemment.
- Méthode de réflectométrie : Cette méthode a le même principe de fonctionnement qu'un radar, une impulsion ou un signal est transmis dans le champ PV à diagnostiquer [8].

1.6.Conclusion

Dans ce chapitre nous avons présenté les générateurs PVs en abordant les défauts qui peuvent arriver lors du fonctionnement :

- Les bulles d'air.
- Le délaminage.
- Le point chaud.
- Le Mis match et l'ombrage partiel.

En intégrant par le défaut de point chaud ; la question qui peut être posée est quelle est la cause d'origine qui provoque ce problème, pour cela on utilise plusieurs méthodes électriques ou non-électriques, telles que le traitement d'image qui va être présenté dans le chapitre suivant.



Chapitre 2

Méthodes de classification d'images



2.1. Introduction

Dans ce chapitre nous sommes intéressés par l'utilisation du traitement d'image pour diagnostiquer les défauts dans un générateur PV, en passant sur les points suivants :

- Traitement d'image.
- Méthodes de classification.
- Apprentissage profond.

2.2. Traitement d'image

2.2.1. Définitions

2.2.1.1. L'image

Une image est une représentation planaire d'un objet quelconque. Mathématiquement, c'est une fonction bidimensionnelle de la forme $f(x, y)$, où $f(x_0, y_0)$ est la valeur de f aux coordonnées (x_0, y_0) . Donc, c'est un processus continu 2D résultant d'une mesure physique. L'amplitude de f est appelée intensité (ou niveau de gris) de l'image au point de coordonnées (x, y) . Si une image est générée par un processus physique, $f(x, y)$ correspond à l'énergie irradiée par ce processus [14].

Quand les valeurs de l'amplitude de f et des coordonnées (x, y) sont des quantités discrètes, l'image est appelée image numérique. Une image numérique est donc composée d'un nombre fini d'éléments, ayant chacun un emplacement et une valeur donnés. Ces éléments sont appelés pixels (contraction de Picture Element) [14].

Ils existent trois types des images :

- Image en niveau de gris ($f(x, y) \in [0..255]$).
- Image binaire ($f(x, y) \in \{0, 1\}$).
- Image couleur ($f_B(x, y) ; f_G(x, y) ; f_R(x, y)$) [15].

2.2.1.2. Traitement d'image

Le traitement d'images est une branche du traitement de signal dédiée aux images et vidéo. Le traitement d'images est l'ensemble des opérations effectuées sur l'image, afin d'en améliorer la lisibilité et d'en faciliter l'interprétation. C'est, par exemple, le cas des opérations de rehaussement de contraste, élimination du bruit et correction d'un flou. C'est aussi l'ensemble d'opérations effectuées pour extraire des "informations" de l'image comme la segmentation et l'extraction de contours [14].

2.2.1.3.L'histogramme

En imagerie, l'histogramme représente la distribution des couleurs au niveau de gris de l'image. C'est un outil fondamental du traitement d'images, avec de très nombreuses applications. Les histogrammes sont aussi très utilisés en photographie et pour la retouche d'images [16].

L'histogramme montre que la couleur noire est présente en abondance dans l'image. Nous pouvons également spécifier le nombre de cases comme deuxième argument utilisé pour calculer l'histogramme. Le nombre de cases doit être un entier positif [17].

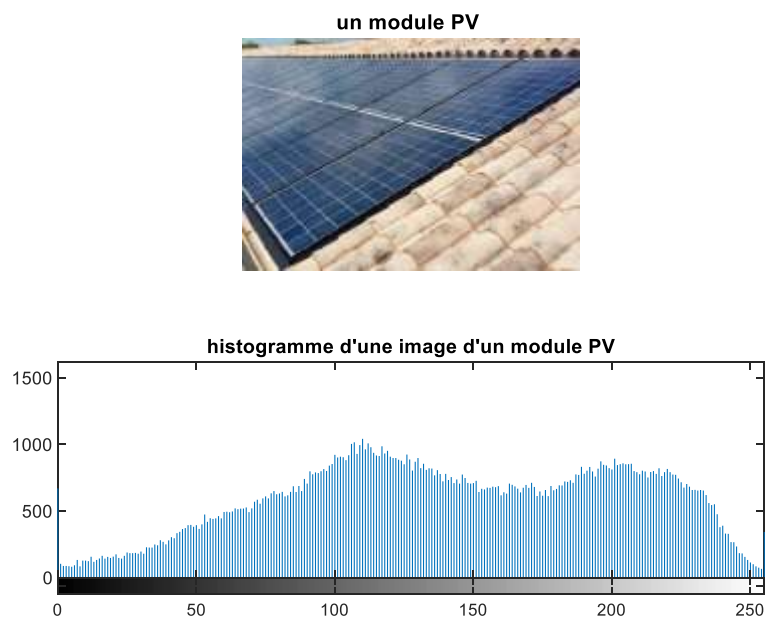


Figure 2.1. Exemple d'un histogramme d'une image fait par Matlab.

2.2.1.4. Quelques techniques de traitement d'image

Il y'a plusieurs méthodes, on cite les suivantes :

- L'amélioration de contraste des images

Le contraste peut être défini de plusieurs façons [15] :

- Ecart – type des variations des niveaux de gris

$$c = \sqrt{\frac{1}{M \times N} \sum_{x=t}^{N-1} \sum_{y=t}^{M-1} (f(x,y) - Moy)^2} \quad (2.1)$$

- Variation entre niveaux de gris

$$c = \frac{\max[f(x,y)] - \min[f(x,y)]}{\max[f(x,y)] + \min[f(x,y)]} \quad (2.2)$$

Ils existent plusieurs méthodes de l'amélioration des images :

- Transformation linéaire.
- Transformation linéaire avec saturation.
- Transformation linéaire par morceau.
- Transformation non-linéaire.
- Egalisation de l'histogramme [15].

- **Détection des contours**

Un contour est un ensemble des points d'une image numérique qui correspond à un changement brutal de l'intensité lumineuse. La détection de contour est une étape incontournable à de nombreuses applications de l'analyse d'images. Car elle permet une réduction importante de la quantité d'information relative à une image, tout en préservant des informations structurelles comme les contours et les frontières des images. Les contours constituent en effet des indices riches, au même titre que les points d'intérêts, pour toute interprétation ultérieure de l'image [18].

- **La segmentation**

La segmentation est l'opération de partition d'une image à des composantes séparées ou à des objets séparés. Son but est de :

- Démarrer l'interprétation de l'image,
- Réduire la quantité de l'information dans l'image, sans nuire de façon sensible à la compréhension,
- Structurer les informations de l'image.

Dans la segmentation d'images, on distingue deux approches :

- Segmentation en régions homogènes (clustering),
- Segmentation par extraction de contours [19].

Il existe principalement trois approches de la segmentation en régions :

- Méthode par séparation (Split).
- Méthode par fusion.
- Méthode par séparation/fusion [19].

On peut ranger ces méthodes en deux catégories :

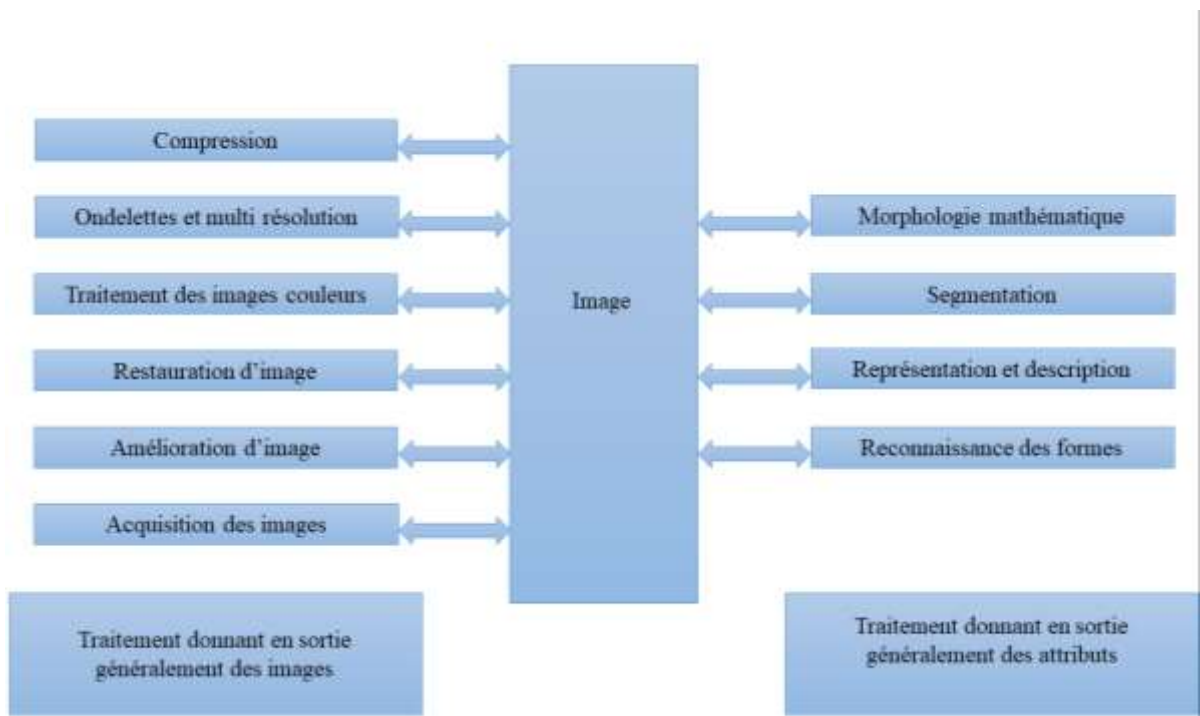


Figure 2.2. Les différents traitements qui peuvent être appliqués sur une image [14].

2.3. Classification des images

La classification est une méthode mathématique utilisée pour l'analyse des données, elle est appliquée sur des données numériques afin de faciliter l'étude d'une population d'effectif important. Classifier une image est une tâche ou une série de méthodes qu'une théorie unifiée est capable d'utiliser des images pour une analyse ultérieure ou pour la cartographie. Il s'agit de déterminer les procédures de collecte de tous les pixels de l'image dans un nombre restreint de classes correspondant aux grands éléments de structure de l'image [20].

On peut classer les méthodes en deux classes principales présentées dans la figure suivante :

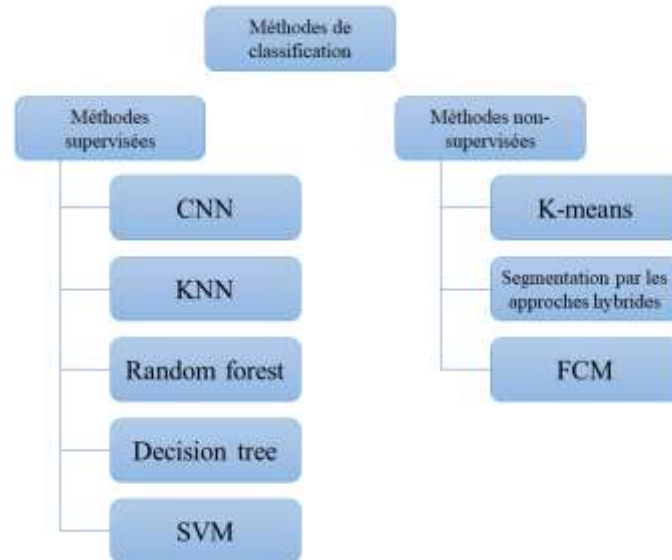


Figure 2.3. Les méthodes de classification des images.

2.3.1. Apprentissage automatique (Machine Learning)

- **K-moyennes (K-means)**

L'algorithme k- moyennes est l'un des algorithmes les plus connus en classification non supervisé, appelée algorithme des centres mobiles. Il partitionne un ensemble de données en un nombre prédéfini de régions K [21].

Chaque groupe est représenté par sa moyenne (centre de la classe) qui ses coordonnées sont la moyenne arithmétique pour chaque dimension séparément de tous les points dans le cluster [21].

- **Floues c-moyennes (FCM)**

Floues C- moyennes (FCM) est une méthode de classification floue non supervisée très populaire, dérivée de l'algorithme c-moyennes, issu de l'algorithme k-moyennes. Elle est fréquemment utilisée dans la reconnaissance des formes. Cet algorithme permet à chaque classe être représentée par son centre de gravité. Ainsi, ce type d'algorithme est fort utilisé en traitement d'images, permet d'obtenir une partition floue de l'image en donnant à chaque pixel un degré d'appartenance à une région donnée [21].

- **K-PV (K-Plus Proche Voisin)**

La méthode du plus proche voisin (k-PPV ou k-NN pour K-Nearest Neighbor) est supervisée et non-paramétrique. L'individu est affecté à la classe qui contient le plus d'individus parmi ces plus proches voisins. Cette méthode exige que l'on choisisse une distance, le plus classique est la distance euclidienne, et le nombre de voisins à tenir compte. En outre, son apprentissage est tout à fait simple, parce qu'il est du type d'apprentissage par cœur (sur garde tous les exemples d'apprentissage). Toutefois, le temps de prévision est très long par ce qu'il faut calculer la distance avec tous les exemples, mais il y a des heuristiques pour réduire le nombre d'exemples à prendre en compte [21].

- **L'arbre de décision (Decision tree)**

Les méthodes d'ensemble d'arbre de décision sont très populaires dans l'apprentissage automatique. Ces méthodes reposent sur l'amélioration d'un algorithme d'apprentissage existant en combinant les prédictions de plusieurs modèles. Ils sont plus efficaces lorsqu'ils sont utilisés avec des arbres de décision qui, autrement, ne sont souvent pas compétitifs en termes de précision avec d'autres algorithmes d'apprentissage. Nous pensons que les méthodes d'ensemble basées sur des arbres de décision constituent un bon point de départ pour concevoir un système générique de classification des images. Ils ne sont pas faits d'hypothèse a priori sur le problème de l'application ; ils ont été appliqués avec succès à de nombreux problèmes complexes dans divers domaines d'application et, ils se comparent très favorablement avec d'autres algorithmes de pointe. Dans l'algorithme d'induction classique, l'arbre est cultivé de façon descendante en utilisant les exemples d'apprentissage et en cherchant à chaque nœud le test qui maximise l'information de mesure du score qui évalue la capacité d'un test à séparer les instances du sous-ensemble d'apprentissage courant [22].

- **Forêt aléatoire (Random forest)**

L'algorithme de forêt aléatoire est une technique d'apprentissage automatique qui est de plus en plus utilisée pour la classification des images et la création de variables continues telles que le pourcentage de couverture arborescente et biomasse forestière. Les forêts aléatoires sont un modèle d'ensemble, ce qui signifie qu'elles utilisent les résultats de nombreux modèles différents pour calculer une réponse. Dans la plupart des cas, le

Le résultat d'un modèle d'ensemble sera meilleur que celui d'un des modèles individuels. Dans le cas des forêts aléatoires, plusieurs arbres de décision sont créés (cultivés) et la réponse est calculée en fonction du résultat de tous les arbres de décision [23].

- **Machine à vecteur de support (Support Vector Machine)**

Les SVM sont des systèmes d'apprentissage qui utilisent un espace d'hypothèse de fonctions linéaires dans un hyperespace, formés avec un algorithme d'apprentissage de la théorie de l'optimisation qui met en œuvre un biais d'apprentissage dérivé de la théorie de l'apprentissage statistique. L'objectif de la classification via SVM est de trouver un moyen informatique efficace d'apprendre à bien séparer les hypers plans dans un hyperespace [22].

- **Réseau de neurones convolutifs (CNN)**

Les neurones sont considérés comme le support physique de l'intelligence. Les réseaux de neurones « artificiels », pour les différencier des réseaux de neurones biologiques, sont composés de neurones artificiels simples, petites fonctions mathématiques, qui permettent, de former des fonctions complexes très utiles [24].

On appelle réseaux de neurones un ensemble d'entités (les neurones) interconnectées. Dans la grande majorité des cas, les neurones sont en fait des fonctions calculées par un programme informatique, mais ils sont parfois réalisés sur des circuits électroniques. Les neurones sont caractérisés par un état d'excitation qui dépend de celui des neurones situés en amont, ainsi que de la force des liens qui les relient [24].

2.3.2. Apprentissage profond (Deep learning)

L'apprentissage en profondeur est un nouveau domaine de recherche du l'apprentissage automatique (ML), qui a été introduit dans le but de rapprocher le ML de son objectif principal : l'intelligence artificielle. Il concerne les algorithmes inspirés par la structure et le fonctionnement du cerveau. Ils peuvent apprendre plusieurs niveaux de représentation dans le but de modéliser des relations complexes entre les données [25].

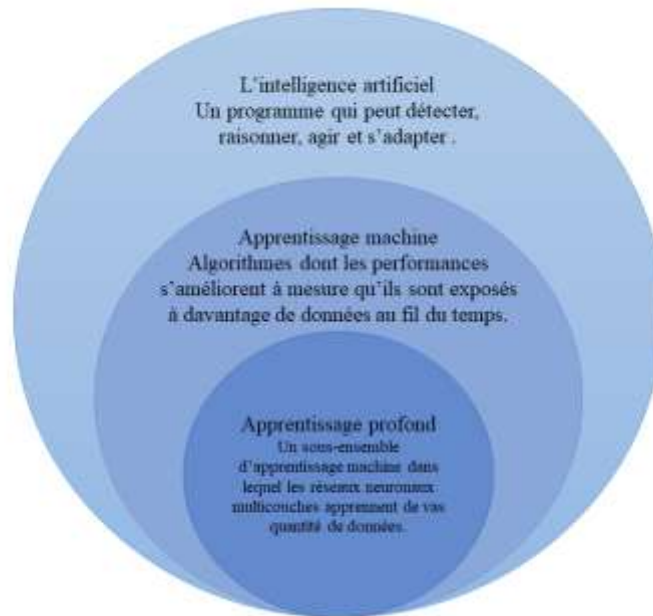


Figure 2.4. La famille de l'apprentissage profond [26].

2.3.2.1. Types d'apprentissage profond

Ils existent trois types de l'apprentissage [26].

- **Apprentissage supervisé profond (deep supervised learning)**

Cette technique traite des données étiquetées. En considérant une telle technique, les environs ont une collection d'entrées et de sorties résultantes $(x_t, y_t) \sim \rho$. Par exemple, l'agent intelligent devine si l'entrée est x_t et obtiendra une valeur de perte. Ensuite, les paramètres du réseau sont mis à jour à plusieurs reprises par l'agent afin d'obtenir une meilleure estimation des extrants privilégiés. Suite à un résultat positif de formation, l'agent acquiert la capacité d'obtenir les bonnes solutions aux requêtes des environs. Pour l'AD, il existe plusieurs techniques d'apprentissage supervisé, telles que les réseaux de neurones récurrents (RNNs), les réseaux de neurones convolutifs (CNNs), et les réseaux de neurones profonds (DNNs). En outre, la catégorie RNN comprend les unités récurrentes synchronisées (SRU) et les approches de mémoire à court terme (LSTM). Le principal avantage de cette technique est la capacité de recueillir des données ou de générer une sortie de données à partir de connaissances antérieures. Cependant, l'inconvénient de cette technique est que les limites de décision peuvent être dépassées lorsque l'ensemble de formation ne possède pas d'échantillons qui devraient être dans une classe. Dans l'ensemble, cette technique est plus simple que d'autres techniques dans la façon d'apprendre avec une haute performance

- **Apprentissage semi-supervisé profond**

Dans cette technique, le processus d'apprentissage est basé sur des ensembles de données semi-marqués. Occasionnellement, les réseaux génératifs antagonistes (GRN) et (DRL) sont utilisés de la même manière que cette technique. De plus, les ARN, qui comprennent les (URG) et les (MTST), sont également utilisés pour l'apprentissage partiellement supervisé. L'un des avantages de cette technique est de minimiser la quantité de données étiquetées nécessaires. D'autre part, l'un des inconvénients de cette technique est les caractéristiques d'entrée non pertinentes présentes dans les données de formation pourrait fournir des décisions incorrectes. Le classificateur de documents textuels est l'un des exemples les plus populaires d'application de l'apprentissage semi-supervisé. En raison de la difficulté d'obtenir un grand nombre de documents textuels étiquetés, l'apprentissage semi-supervisé est idéal pour les tâches de classification des documents textuels.

- **Apprentissage non-supervisé profond**

Cette technique permet de mettre en œuvre le processus d'apprentissage en l'absence des données étiquetées disponibles (c.-à-d. qu'aucune étiquette n'est requise). Ici, l'agent apprend les caractéristiques significatives ou la représentation intérieure nécessaire pour découvrir le non identifié structure ou relations dans les données d'entrée. Techniques de réseaux génératifs, réduction de la dimensionnalité et regroupement sont souvent comptés dans la catégorie d'apprentissage non supervisé. Plusieurs membres de la famille DL ont la réduction non linéaire de la dimensionnalité et les tâches de regroupement, notamment les tâches restreintes les machines Boltzmann, les auto-encodeurs et les (GNA) sont les techniques les plus récentes. En outre, l'apprentissage non supervisé dans un large éventail d'applications. Les inconvénients de l'apprentissage non supervisé sont incapables de fournir des renseignements exacts concernant le tri des données et sont complexes. L'un des plus populaires approches d'apprentissage non supervisées se regroupent.

- **Apprentissage profond par renforcement**

Renforcement d'apprentissage repose sur l'interaction avec l'environnement, tandis que l'apprentissage supervisé repose sur des données d'échantillons fournies. Cette technique a été développée en 2013 avec Google Deep Mind. Par la suite, de nombreuses techniques améliorées dépendant de l'apprentissage par renforcement ont été construites. Par

exemple, si les échantillons d'environnement d'entrée : $x_t \sim \rho$, agent prédire $y^*_t = x(f_t)$: et le coût reçu de l'agent est $c_t \sim P(c_t | x_t, y^*_t)$, P ici est la distribution de probabilité inconnue, puis l'environnement pose une question à l'agent. La réponse qu'il donne est un score bruyant. Cette méthode est parfois appelée apprentissage semi-supervisé. Sur la base de ce concept, plusieurs techniques supervisées et non supervisées ont été élaborées. En comparaison avec les techniques traditionnelles supervisées, l'exécution de cet apprentissage est beaucoup plus difficile, car aucune fonction de perte directe n'est disponible dans la technique d'apprentissage de renforcement. En outre, il existe deux différences essentielles entre l'apprentissage supervisé et l'apprentissage par renforcement : premièrement, il n'y a pas d'accès complet à la fonction, ce qui nécessite une optimisation, ce qui signifie qu'elle doit être interrogée par interaction ; deuxièmement, l'état avec lequel on interagit est fondé sur un environnement, où l'input x_t est basé sur les actions précédentes

2.3.2.2. Réseaux de neurone convolutifs profonds

i. Historique

Les réseaux de neurones convolutifs ne viennent pas du rien, mais sont été développés durant les années passées [27].

- **Le neurone formel**

En 1943 Warren McCulloch et Walter Pitts publient leur premier modèle mathématique et informatique du neurone biologique : le neurone formel.

Le neurone formel est inspiré directement du neurone biologique, Son fonctionnement est simple, le neurone active sa sortie (sortie active = 1) selon si ses entrées dépassent un certain seuil. À partir de ce neurone, il est possible de construire n'importe quel réseau de neurone artificiel capable de résoudre des opérations logiques.

- **Le perceptron**

Il est inventé en 1957 par F. Rosenblatt. Le perceptron est un neurone formel, dont la fonction d'activation est une fonction échelon (Linear Threshold Function), ce qui fait du perceptron une unité linéaire à seuil (ou Linear Threshold Unit). Ses entrées sont des nombres quelconques, et chaque entrée est pondérée par un poids (w).

- **Le perceptron multicouche**

Il a été inventé en 1986, et généralement se présente sur la forme suivante :

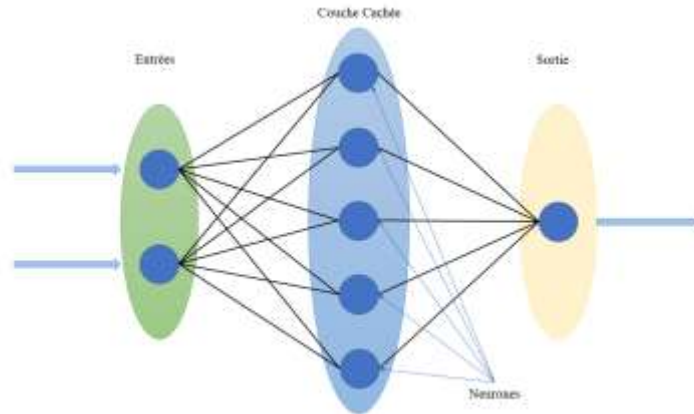


Figure 2.5. Perceptrons multicouches.

On distingue trois couches :

- Couche d'entrée.
- Couches cachées (ici une seule couche avec cinq neurones).
- Couche de sortie.

Les perceptrons multicouches sont des réseaux neuronaux et ont été réalisés afin de classer des données plus complexes que celles classées par un perceptron [27].

- **Réseaux de neurone convolutifs (CNN)**

Les réseaux de neurones convolutifs constituent un type de réseau de neurones spécialisés pour le traitement de données. Jusqu'à présent, il s'agit des modèles de classement les plus efficaces. Désignés par l'acronyme CNN, de l'anglais Convolutional Neural Network.

Comme entrée, une image est fournie sous la forme d'une matrice de pixels. Il a deux dimensions pour une image en niveaux de gris. La couleur est représentée par une tierce dimension, profondeur 3 pour représenter les couleurs fondamentales [Rouge, Vert, Bleu] [28].

- **Réseaux de neurone convolutifs profonds (DCNN)**

Un réseau de neurones profond est un réseau de neurone avec au minimum deux couches cachées. Les réseaux de neurones convolutifs profonds ont été mis en utilisation à 2010 [27].

Ces réseaux de neurones multicouches peuvent comporter des millions de neurones, répartis en plusieurs dizaines de couches. Ils sont utilisés en apprentissage profond pour concevoir des mécanismes d'apprentissage supervisés et non supervisés [25].

ii. Architecture de base

Les DCNN comportent deux parties bien distinctes. La première partie d'un CNN est la partie convolutive elle-même. Elle fonctionne d'extracteur des caractéristiques de l'image. Une image est passée par une succession de filtres, ou noyaux de convolution, créant de nouvelles images des cartes de convolution. Certains filtres intermédiaires réduisent la résolution l'image par une opération locale maximale. En fin de compte, les cartes de convolution sont plates et concaténées en un vecteur de caractéristiques, appelé code CNN [28].

Ce code CNN en sortie de la partie convolutive est ensuite branché en entrée d'une deuxième partie, constituée de couches entièrement connectées (perceptron multicouche), Le rôle de cette partie est de combiner les caractéristiques du code CNN pour classer l'image. La sortie est une dernière couche comportant un neurone par catégorie. Les valeurs numériques obtenues sont généralement normalisées entre 0 et 1, de somme 1, pour produire une distribution de probabilité sur les catégories [28]. L'architecture de base d'un CNN est présenté dans la figure suivante :

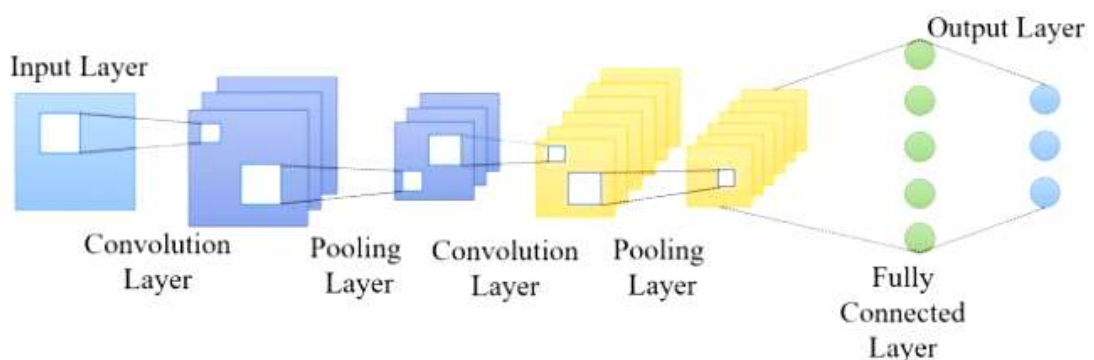


Figure 2.6. Architecture de base [29].

iii. Les couches principales d'un DCNN

Comme montrée dans la figure 2.6 ; un réseau de neurone est composé de cinq couches de base : la couche d'entrée, la couche de convolution, la couche de pooling, la couche entièrement connectée et la couche de sortie ; plus la fonction d'activation, la couche de correction (ReLU) et de perte (loss).

- **Couche d'entrée (Input layer) :** L'ensemble de données brutes d'entrée peut être directement entré dans la couche d'entrée. Une image est entrée par sa valeur de pixel dans la couche d'entrée [29].
- **Couche de convolution (CONV) :** Aussi connue par le nom *non up-sampling layer* [30], cette couche est la composante clé des réseaux de neurones convolutifs, et constitue toujours au moins leur première couche. Son but est de repérer la présence d'un ensemble des caractéristiques dans les images reçues en entrée. Pour cela, on réalise un filtrage par convolution : le principe est de faire "glisser" une fenêtre représentant la caractéristique sur l'image et de calculer le produit de convolution entre la caractéristique et chaque portion de l'image balayée. Une caractéristique est alors vue comme un filtre : les deux termes sont équivalents dans ce contexte [28].
- **La fonction d'activation :** Afin d'apprendre l'approximation universelle des valeurs d'entrée et des classes de sortie comme une fonction continue dans l'espace euclidien, une fonction d'activation appropriée est essentielle. Comme le problème des fonctions tangentes sigmoïde et hyperbolique de gradient de disparition, une unité rectiligne est utilisée comme une fonction d'activation et retourne la même valeur (x) des cartes de caractéristiques fournies en sortie (x) si sa magnitude est supérieure à zéro [30].
- **Couche de pooling (POOL) :** Aussi connue par le nom *down-sampling layer* [29], le pooling est une forme de sous-échantillonnage de l'image. L'image d'entrée est découpée en une série de rectangles de n pixels sur le côté qui ne se chevauchent pas (mise en commun). Chaque rectangle peut être vu comme une tuile, Le signal de la tuile de sortie est défini en fonction des valeurs prises par les différents carreaux pixels [28].

Il y'en a plusieurs types, on cite ces deux types :

AveragePooling

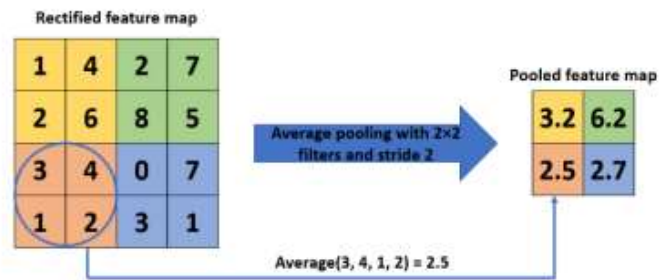


Figure 2.7. Exemple d'une opération AveragePooling [31].

MaxPooling

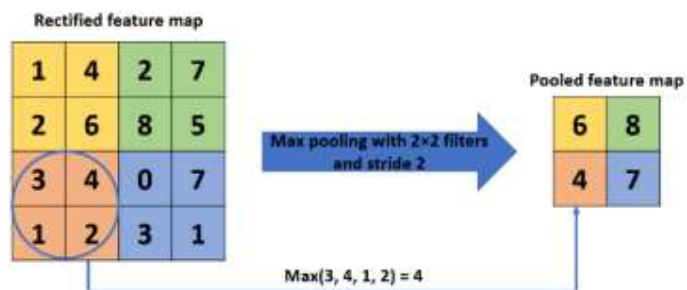


Figure 2.8. Exemple d'une opération MaxPooling [31].

- **Couche de correction (ReLU) :** Pour améliorer l'efficacité du traitement en intercalant entre les couches de traitement une couche qui va opérer une fonction mathématique (fonction d'activation) sur les signaux de sortie. Dans ce cadre on trouve ReLU (Rectified Linear Units) désigne la fonction réelle non-linéaire définie par :

$$\text{ReLU}(x) = \max(0, x) \quad (2.3)$$

La couche de correction ReLU remplace donc toutes les valeurs négatives reçues en entrées par des zéros. En ne modifiant pas les données positives, ReLU n'impacte pas les caractéristiques mises en évidence par la convolution, au contraire : elle les met davantage en évidence en creusant l'écart (valeurs négatives) « entre » deux caractéristiques [28]

- **Couche entièrement connectée (Fully connected layer FC) :** Les cartes des caractéristiques de résultat de la couche de convolutions, pooling et ReLU fournissent au final un ensemble de caractéristiques données sous forme de carte 2D. Ces cartes sont concaténées en un vecteur de caractéristiques, appelé

code CNN. Ce code CNN en sortie de la partie convolutive est ensuite branché en entrée d'une deuxième partie, constituée d'une ou de plusieurs couches entièrement connectées (FC ou fully connected) qu'on peut assimiler au perceptron multicouche. Les neurones dans une couche entièrement connectée ont des connexions vers les sorties de la couche précédente. Leurs fonctions d'activation peuvent donc être calculées avec une multiplication matricielle entre le vecteur d'entrée H et la matrice des poids F auquel on ajoute le vecteur des biais des neurones C :

$$A = FH + C \quad (2.4)$$

Le nombre de neurones de la dernière couche entièrement connectée est égal au nombre de classes. Le rôle de cette partie est donc de combiner les caractéristiques du code CNN pour classer une image [28].

- **Couche de perte (loss) :** La couche de perte spécifie comment l'entraînement du réseau pénalise l'écart entre le signal prévu et réel. Elle est normalement la dernière couche dans le réseau. Diverses fonctions de perte adaptées à différentes tâches peuvent y être utilisées. La fonction « Softmax » permet de calculer la distribution de probabilités sur les classes de sortie [28].
- **Couche de sortie (output layer) :** Le nombre de neurones dans cette couche est défini selon les conditions requises. Si la classification est requise, le nombre de neurones est généralement lié au nombre de catégories à classer [30].

iv. Exemples des DCNN les plus utilisés

On peut résumer les CNN connus dans le tableau suivant :

Tableau 2.1. Bref aperçu des architectures CNN [26].

Modèle	Conclusion principale	Pronfond	Base de données	Erreur	Taille de l'entrée	Année
AlexNet	Utiliser Dropout et ReLU	8	ImageNet	16.4	277x277x3	2012
MIN	Prochaine couche, nommée 'mlpcon', utiliser GAP	3	CIFAR-10, CIFAR 100, MNIST	10.41, 35.68, 0.45	32x32x3	2013

ZfNet	Idée de visualisation des couches au milieu	8	ImageNet	11.7	224x224x3	2015
VGG	Augmenter le profond Petite taille du filtre	16,19	ImageNet	7.3	224x224x3	2014
GoogleNet	Profondeur accrue, concept de bloc, taille de filtre différente, concept de concaténation	48	ImageNet	6.7	224x224x3	2015
Inception-V3	Utilise une petite taille de filter, une meilleure représentation des fonctionnalités	48	ImageNet	3.5	229 × 229 × 3	2015
Highway	Présentation du concept de parcours multiples	19, 32	CIFAR-10	7.76	32 × 32 × 3	2015
Inception-V4	Concepts de transformation et d'intégration divisés	70	ImageNet	3.08	229 × 229 × 3	2016
ResNet	Robuste contre le débordement en raison de liens de saut basés sur la symétrie	152	ImageNet	3.57	224 × 224 × 3	2016
Inception-ResNet v2	Introduction du concept de liens résiduels	164	ImageNet	3.52	229 × 229 × 3	2016
FractalNet	Introduit le concept de Drop-Path comme régularisation	40,80	CIFAR-10 CIFAR-100	4.60 18.85	32 × 32 × 3	2016
WideResNet	Diminution de la profondeur et augmentation de la largeur	28	CIFAR-10 CIFAR-100	3.89 18.85	32 × 32 × 3	2016
Xception	Une convolution dépthwise suivie d'une convolution pointwise	71	ImageNet	0.055	229 × 229 × 3	2017
Residual attention	Présentation de la technique d'attention	452	CIFAR-10 CIFAR-100	3.90, 20.4	40x40x3	2017

neural network						
Squeeze-and-excitation networks	Interdépendances modélisées entre les canaux	152	ImageNet	2.25	229 × 229 × 3 224 × 224 × 3 320 × 320 × 3	2017
DenseNet	Blocks of layers; layers connected to each other	201	CIFAR-10, CIFAR100, ImageNet	3.46, 17.18, 5.54	224 × 224 × 3	2017
Competitive squeeze and excitation network	Blocs de couches ; couches reliées entre elles	152	CIFAR-10, CIFAR100,	3.58, 18.47	32 × 32 × 3	2018
MobileNet-v2	Structure résiduelle inversée	53	ImageNet	-	224 × 224 × 3	2018
CapsuleNet	Fait attention aux relations particulières entre les caractéristiques	3	MNIST	0.00855	28 × 28 × 1	2018
HRNetV2	Représentations haute résolution	-	ImageNet	5.4	224 × 224 × 3	2020

2.4. Conclusion

Dans ce chapitre nous avons élaboré un bref aperçu sur les CNNs et leur utilisation pour la classification des images, cette méthode que nous avons utilisée pour classer les défauts d'origine qui provoque les points chauds et ce qui seront montrés dans le chapitre suivant.



Chapitre 3

Mise en œuvre d'une méthode de diagnostic des générateurs photovoltaïques



3.1. Introduction

Rappelons que notre objective est de réaliser un système semi-automatique pour le diagnostic d'un système PV. Il s'agit de détecter le défaut s'il existe dans un générateur PV, puis le classer selon des classes connues.

Dans ce chapitre nous exposons ce travail en abordant les points suivants :

- Ressources (matériels et logiciels).
- Application de l'apprentissage profond (DL) pour le diagnostic des générateurs PVs.
- Mise en œuvre de la méthode.

Les codes de base utilisés dans ce chapitre sont extraits de la référence suivante [32].

3.2. Ressources

3.2.1. Ressources matérielles

Notre système est développé dans un laptop dont les caractéristiques techniques, sont les suivantes :

- Fabricant du système : DELL.
- Modèle de système : DESKTOP-9A6QDQ4.
- Processeur: Intel(R) Core (TM) i5-6200U CPU @ 2.30GHz 2.40 GHz.
- RAM: 8 Go (7.89 Go utilisable).

3.2.2. Ressources logicielles

- Système d'exploitation

Les codes ont été développés sur Windows 10 Professionnel, puis implémentés sur une carte Raspberry PI 4 en utilisant Raspbian.

- Langage de programmation

Les codes utilisés sont développés en utilisant Python, sur Google Colab et Anaconda. On a utilisé Python 3.10.1.

- **Python** : est un langage de programmation de haut niveau interprété (il n'y a pas d'étape de compilation) et orienté objet avec une sémantique dynamique. Il est très sollicité par une large communauté de développeurs et de programmeurs. Python est un langage simple, facile à apprendre et permet une bonne réduction du coût de la maintenance des codes. Les bibliothèques (packages) python

encouragent la modularité et la réutilisabilité des codes. Python et ses bibliothèques sont disponibles sans charges pour la majorité des plateformes et peuvent être redistribués gratuitement [25].

- **Google Colab** : Google Colab ou Colaboratory est un service cloud, offert par Google (gratuit), basé sur Jupyter Notebook et destiné à la formation et à la recherche dans l'apprentissage automatique. Cette plateforme permet d'entraîner des modèles d'apprentissage automatique directement dans le cloud. Sans donc avoir besoin d'installer quoi que ce soit sur notre ordinateur à l'exception d'un navigateur [33].

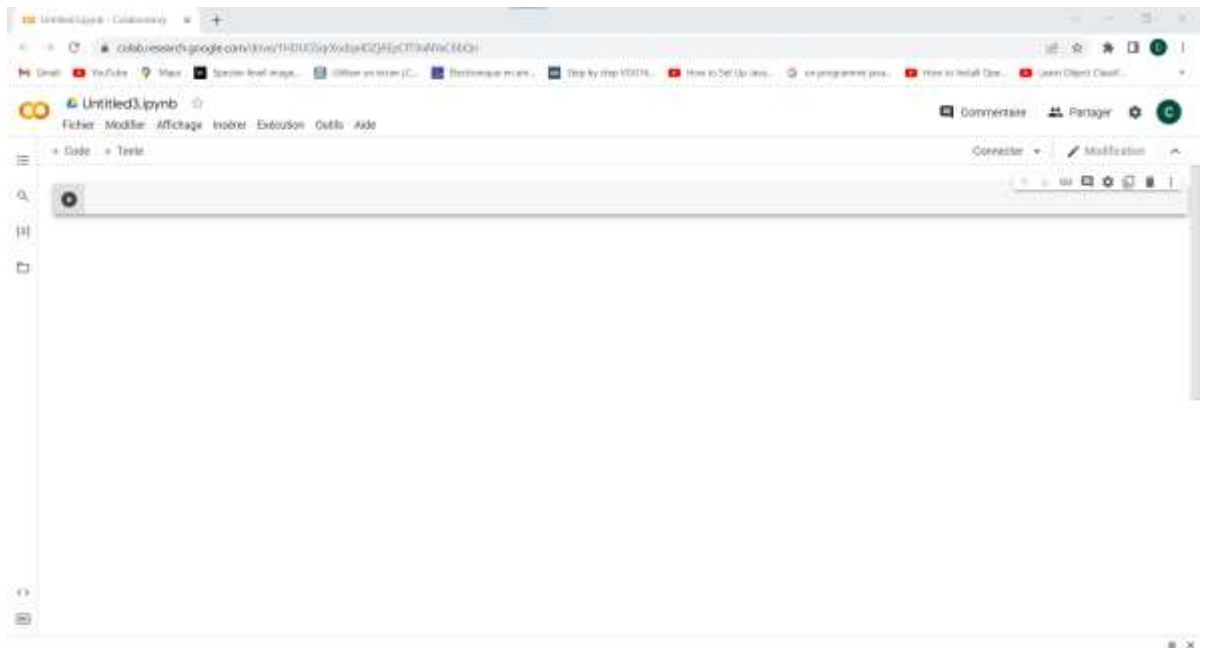


Figure 3.1. Une page de coder sur Colab.

- **Anaconda** : Est une distribution libre et open source des langages de programmation Python et R appliqué au développement d'applications dédiées à la science des données et à l'apprentissage automatique (traitement de données à grande échelle, analyse prédictive, calcul scientifique), qui vise à simplifier la gestion des paquets et de déploiement. Les versions de paquetages sont gérées par le système de gestion de paquets Conda. La distribution Anaconda est utilisée par plus de 20 millions d'utilisateurs, plus de 7500 packages de science des données et d'apprentissage automatique adaptés pour Windows, Linux et MacOS [25].

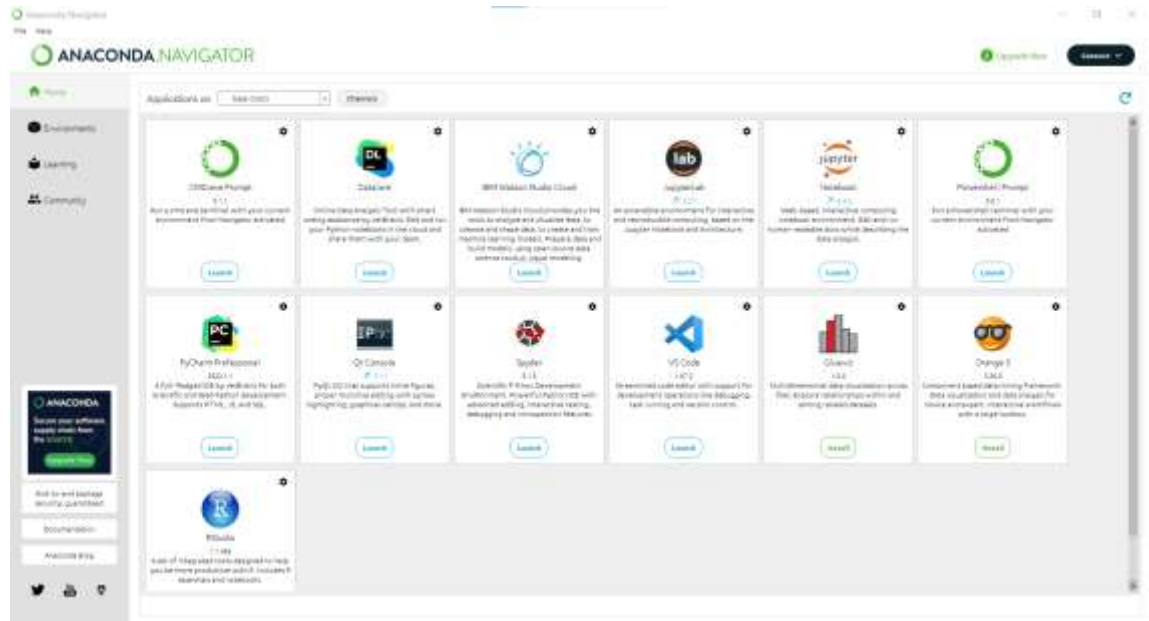


Figure 3.2. Interface Anaconda.

Nous avons aussi installé Tensorflow et Keras pour les applications de la classification.

- **Tensorflow** : C'est une bibliothèque open-source de logiciels pour le flux de données et la programmation différentielle pour diverses tâches. De même, TensorFlow est utilisé dans l'apprentissage automatique par les réseaux de neurones. Développé par Google en 2011 sous le nom de DistBelief, TensorFlow a été officiellement lancé en 2017 gratuitement. La bibliothèque est capable de fonctionner sur plusieurs CPUs et GPUs et est disponible sur plusieurs plateformes, y compris mobiles. Le nome vient de tableaux multidimensionnels appelés tenseurs, couramment utilisés dans les réseaux de neurones [25].
- **Keras** : C'est une API (application programming interface) de réseaux de neurones de haut niveau, écrite en Python et capable de fonctionner sur TensorFlow ou Theano. Il a été développé en mettant l'accent sur l'expérimentation rapide. Être capable d'aller de l'idée à un résultat avec le moins de délai possible est la clé pour faire de bonnes recherches. Il a été développé dans le cadre de l'effort de recherche du projet ONEIROS (Open-ended Neuro-Electronic Intelligent Robot Operating System), et son principal auteur et mainteneur est François Chollet, un ingénieur Google [25].

3.3. Application de l'apprentissage profond (DL) pour le diagnostic des générateurs PVs

Le tableau suivant contient un bref résumé sur les travaux d'application de l'apprentissage profond pour le diagnostic des générateurs photovoltaïques.

Tableau 3.1. Applications de DL dans le diagnostic des générateurs PV du 2018 à 2020 [34].

Année	Méthode basée sur DL	Identification, détection & localisation	Système PV	Complexité, intégration et logiciels	Type de défaut	Proto-type développement & datasets	Vérification en temps réel	Taux & cost
2018	DCNN	Localisation & classification	LSPV farm	Open source relativement complexe (Python)	Élimination, protection contre la poussière, traînées d'escargots, jaunissement	Non 7560 images	Oui/ Online	96%–100% à coût élevé
2018	CNN (VGG-16)	Détection	LSPVP	Environnement de test complexe et strict Code open-source	Anomalies	Non 9732 images	Non/ Offline	– à coût élevé
2019	CNN-MC-SVM	Détection & Identification	LSPVP	Opération complexe Op en code source	élimination, ombrage, traînées d'escargots, jaunissement	Non –	Oui /Online	95.03%–98.42% à coût élevé
2019	DCGAN & CNN	Identification	Chaîne PV	Open source relativement complexe (Python)	Défaut d'Arc	Non 30,000 samples	Non/ Offline	97.68% à coût élevé
2019	LSTM	Classification	Réseau PV	Source ouverte moyenne	LL HSp	Non 6067 échantillons	Non/ Offline	98% à faible coût
2019	DCNN	Détection	Module PV	Open source simple (Python)	Défaut sur les cellules solaires	Non 1000 images IL	Oui/ Online	93.02% Moyen
2019	DCNN	Classification	Chaîne PV	Medium Open source (Python)	LL & OC	No n Signal 2D du courant et voltage	Non/Online	99.51% –
2019	RestNet	Détection & identification	Module PV	Relativement complexe Matlab/Simulink	SC, OC, PSh & dégradation	No n Courbe I–V	Non/ Offline	99.94% à faible coût

2020	DCNN	Détection	Module PV	Medium Open source (Python)	Défaut sur les cellules solaires	Non Images IR	Oui /Online	99.23% Moyen
------	------	-----------	-----------	-----------------------------	----------------------------------	---------------	-------------	-----------------

3.4. Méthodologie

3.4.1. Description

Dans ce qui suit nous allons suivre les étapes présentées dans la figure 3.3.

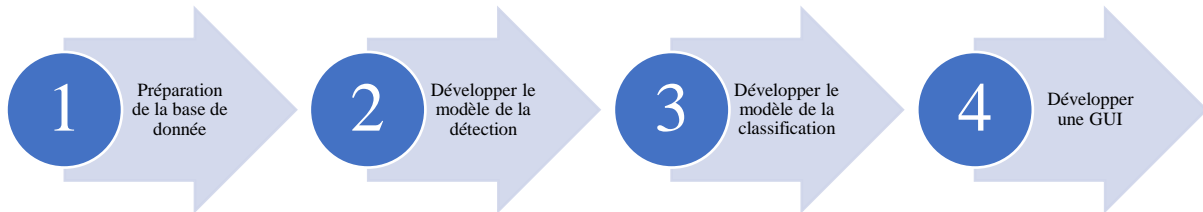


Figure 3.3. Les étapes de travaille.

3.4.2. Base de données utilisée

Notre base de données contient des images thermiques infrarouge afin de faire l'apprentissage et le test de notre système PV. Ces images ont été prises par une caméra thermique associée à un drone qui survole un système photovoltaïque, préparées et prétraitées au niveau du Centre de recherche, UDES-CDER, Bousmail, Tipaza, par Mr. Kellile Nouamen (co-directeur du mémoire), le système est montré dans la figure 3.4.a, la caméra Caméra thermique professionnelle FLIR T540, 24° utilisée est présentée dans la figure 3.4.b.



(a)



(b)

Figure 3.4. Système PV et caméra FLIR [32]

Nous avons organisé cette base de données en deux parties stockées dans deux dossiers qui seront ensuite importées à google drive afin de faciliter l'accès du Google Colab aux données ; la première est pour la détection des défauts contenant des images (335 Mo) appartiennent deux classes : normale et abimée (faulty), voir figure 3.5.

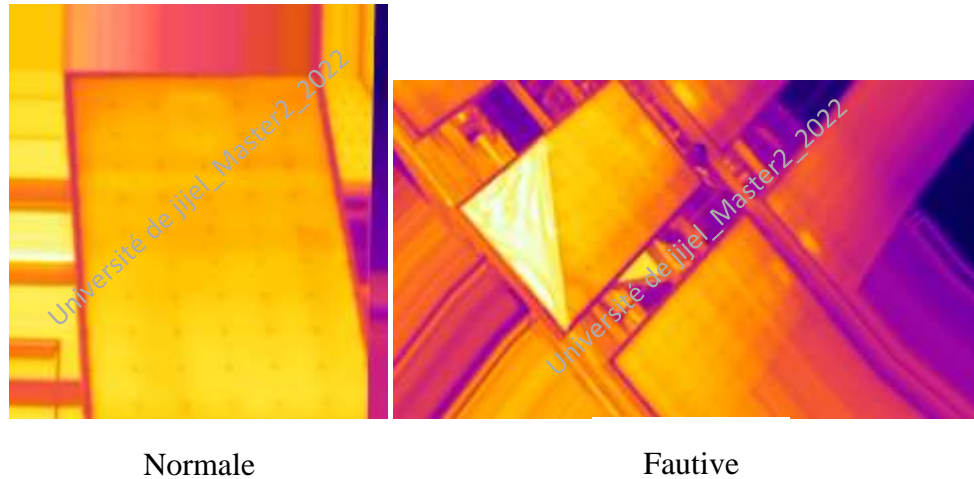


Figure 3.5. Base de données pour la détection.

La deuxième partie (281 Mo) images en séparant en quatre classes conviennent aux quatre défauts :

BD : Diode bypass endommagée.

Du : Dépôt de poussière sur le module PV.

SC : Module PV court-circuité.

SD : L'ombrage partiel.

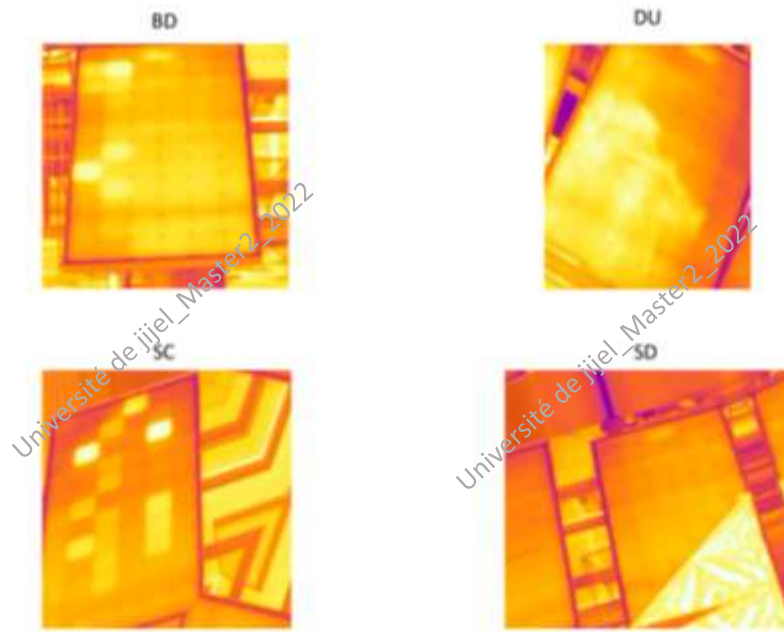
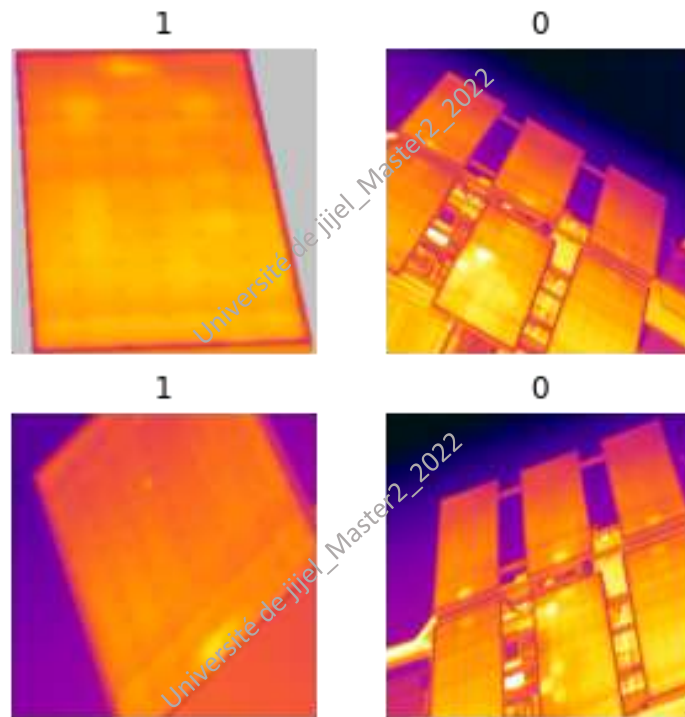


Figure 3.6. Base de données pour la classification.

3.4.3. Développement du deux modèles

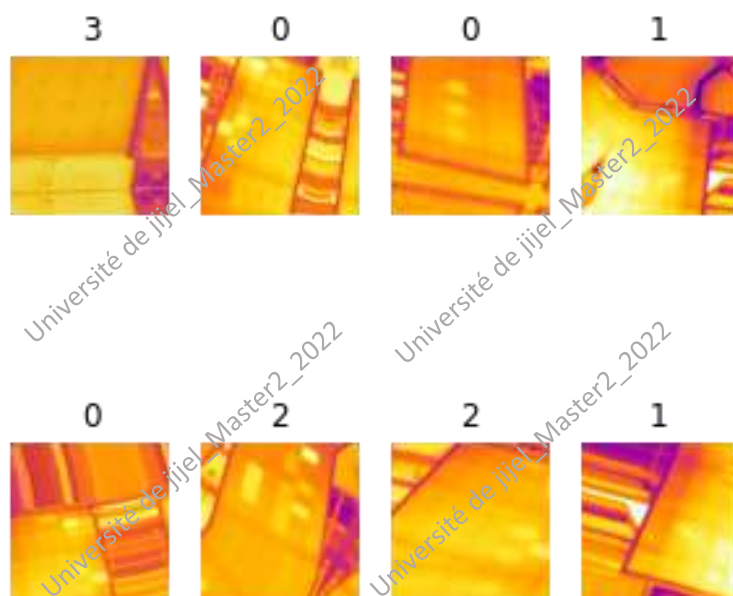
Notre code Python, fait d'abord importer les images du Google Drive cette base de données contient 2008 images, pour la détection ; on a utilisé 80% de images pour faire l'apprentissage et 20% pour le test, cependant la classification on a utilisé 67% pour l'apprentissage et 33% pour le test, durant l'apprentissage le système prend des échantillons aléatoires à partir de la base de données. Les exemples de ces échantillons sont présentés dans la figure 3.7.a et la figure 3.7.b pour la détection et la classification respectivement.



(a)

0 : il y'a un défaut dans le générateur (système endommagé).

1 : le générateur est en bon état.



(b)

Figure 3.7. Echantillons aléatoires

0 : BD

1 : Du

2 : SC

3 : SD

3.4.4. Architecture

Nous avons utilisé la même architecture pour les deux modèles avec 100 epochs pour l'apprentissage en optimisant avec Adamax, elles nous donnent de différents nombres de paramètres ; dans les deux tableaux suivants on présente cette architecture et le nombre de paramètres de chaque modèle.

Tableau 3.2. Architecture et paramètres du modèle de la détection.

Couche	Taille de la sortie	Nombre des paramètres
Average_pooling2d_1 (AveragePooling 2D)	(None,84,84,3)	0
Conv2d_3(Conv2D)	(None,84,84,128)	512
max_pooling2d_3 (MaxPooling2D)	(None, 42, 42, 128)	0
conv2d_4 (Conv2D)	(None, 42, 42, 64)	8256
max_pooling2d_4 (MaxPooling2D)	(None, 21, 21, 64)	0
conv2d_5 (Conv2D)	(None, 20, 20, 32)	8224
max_pooling2d_5 (MaxPooling 2D)	(None, 10, 10, 32)	0
dropout_1 (Dropout)	(None, 10, 10, 32)	0
flatten_1 (Flatten)	(None, 3200)	0
dense_4 (Dense)	(None, 128)	409728
dense_5 (Dense)	(None, 64)	8256
dense_6 (Dense)	(None, 8)	520
dense_7 (Dense)	(None, 2)	18
Total params	435,514	
Trainable params	435,514	
Non-trainable params	0	

Tableau 3.3. Architecture et paramètres du modèle de la classification.

Couche	Taille de la sortie	Nombre des paramètres
Average_pooling2d_1 (AveragePooling 2D)	(None,84,84,3)	0
Conv2d_3(Conv2D)	(None,84,84,128)	512
max_pooling2d_3 (MaxPooling2D)	(None, 42, 42, 128)	0
conv2d_4 (Conv2D)	(None, 42, 42, 64)	8256
max_pooling2d_4 (MaxPooling2D)	(None, 21, 21, 64)	0
conv2d_5 (Conv2D)	(None, 20, 20, 32)	8224
max_pooling2d_5 (MaxPooling 2D)	(None, 10, 10, 32)	0
dropout_1 (Dropout)	(None, 10, 10, 32)	0
flatten_1 (Flatten)	(None, 3200)	0
dense_4 (Dense)	(None, 128)	409728
dense_5 (Dense)	(None, 64)	8256
dense_6 (Dense)	(None, 8)	520
dense_7 (Dense)	(None, 2)	36
Total params	435,532	
Trainable params	435,532	
Non-trainable params	0	

3.4.5. Développement de l'interface graphique

Le programme principal

Le programme principal contient les deux modèles sauvegardés, et les fonctions de l'affichage (envoi du mail et sms). On résume ce programme dans l'organigramme dans la page suivante :

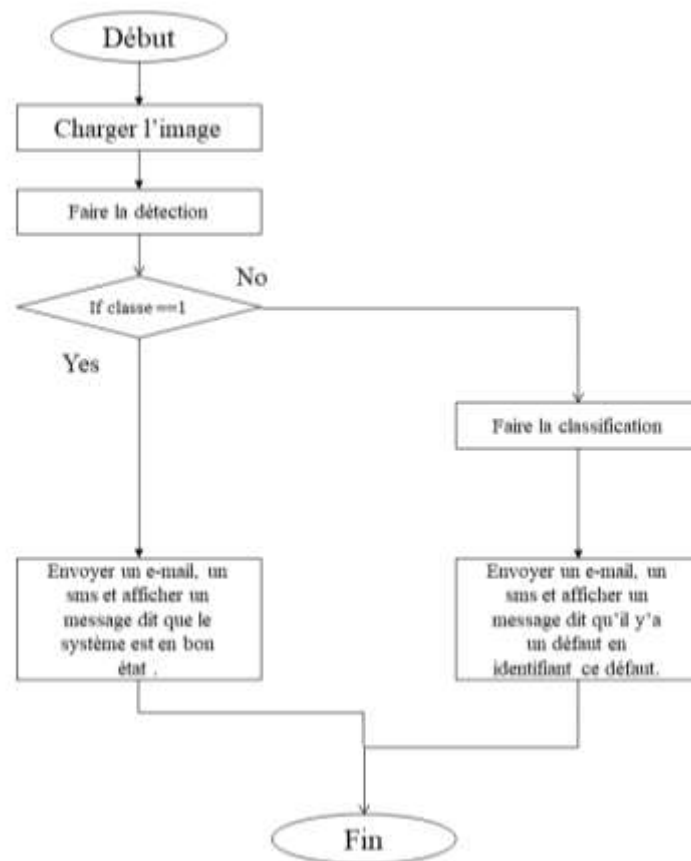


Figure 3.8. Organigramme du fonctionnement.

Nous avons développé une simple interface graphique en utilisant **Tkinter** pour offrir à l'utilisateur de faire afficher les résultats d'une manière simple, en donnant la main de choisir la source de son image ; puis détecter les défauts et le classifier s'il y'en a.

3.5. Conclusion

Dans ce chapitre, nous avons parlé brièvement de la préparation de nos modèles de classification, en passant par la présentation de la base de données, l'apprentissage et finalement la création de l'interface d'affichage graphique. Les résultats obtenus seront présentés dans le chapitre suivant.



Chapitre 4

Test et résultats expérimentaux



4.1. Introduction

Dans ce chapitre, nous allons présenter les résultats de la simulation ainsi que le test pratique. En énumérant :

- Résultats d'apprentissage et validation.
- Résultats du test et simulation.
- Utilisation de l'interface graphique.

4.2. Méthodes d'évaluation

4.2.1. Matrice de confusion

Pour évaluer les résultats obtenus nous avons utilisé la matrice de confusion qu'est un outil de mesure de la performance des modèles de classification à 2 classes ou plus [35].

La matrice de confusion est composée de 4 valeurs comme montré dans la figure 4.1.

Matrice de confusion		Classe prédise	
		Positive	Négative
Classe réelle	Positive	TP (True Positive)	FN (False Negative)
	Négative	FP (False Positive)	TN (True Negative)

(a) La matrice de confusion de la classification binaire.

Matrice de confusion	Classe 1	Classe 2	Classe 3	Classe 4
Classe 1	Cellule 1	Cellule 2	Cellule 3	Cellule 4
Classe 2	Cellule 5	Cellule 6	Cellule 7	Cellule 8
Classe 3	Cellule 9	Cellule 10	Cellule 11	Cellule 12
Classe 4	Cellule 13	Cellule 14	Cellule 15	Cellule 16

(b) La matrice de confusion de la multi-classification (4 classes).

Figure 4.1. Format de la matrice de confusion.

$$\text{Précision} = \frac{TP}{TP+FO} \quad (4.1)$$

$$\text{Rappel} = \frac{TP}{TP+FN} \quad (4.2)$$

$$\text{F1-Score} = \frac{TP}{TP + \frac{1}{2}(FN+FP)} = \frac{2}{\frac{1}{\text{précision}} + \frac{1}{\text{rappel}}} \quad (4.3)$$

4.2.2. Courbe AUC-ROC (Area Under Curve-Receiver Operating Characteristic)

La courbe des caractéristiques de l'opérateur récepteur (ROC) est une mesure d'évaluation des problèmes de classification binaire. Il s'agit d'une courbe de probabilité qui trace le TPR par rapport au FPR à différentes valeurs de seuil et sépare essentiellement le « signal » du « bruit ». L'aire sous la courbe (AUC) est la mesure de la capacité d'un classificateur à distinguer les classes et sert de résumé de la courbe ROC. La courbe ROC plote deux paramètres [36] :

$$\text{TPR} = \frac{TP}{TP+FN} \quad (4.4)$$

$$\text{FPR} = \frac{FP}{FP+TN} \quad (4.5)$$

4.3. Matériel utilisé

Afin de réaliser le système de détection et classification, nous avons utilisé un Raspberry PI 4 où les codes Python ont été intégrés, une carte SIM où le Raspbian été installé et un module SIM808-GSM/GPRS pour l'envoi du SMS.

4.4. Résultats d'apprentissage et validation

4.4.1. La détection

4.4.1.1. Précision d'apprentissage et validation

Ce modèle est un classificateur binaire, qui donne deux classes 0 et 1, en faisant l'apprentissage on obtient ces courbes de changement de précision et la perte en fonction des époques données. La figure 4.2 suivante montre que notre système apprend environ 99% de l'information données.

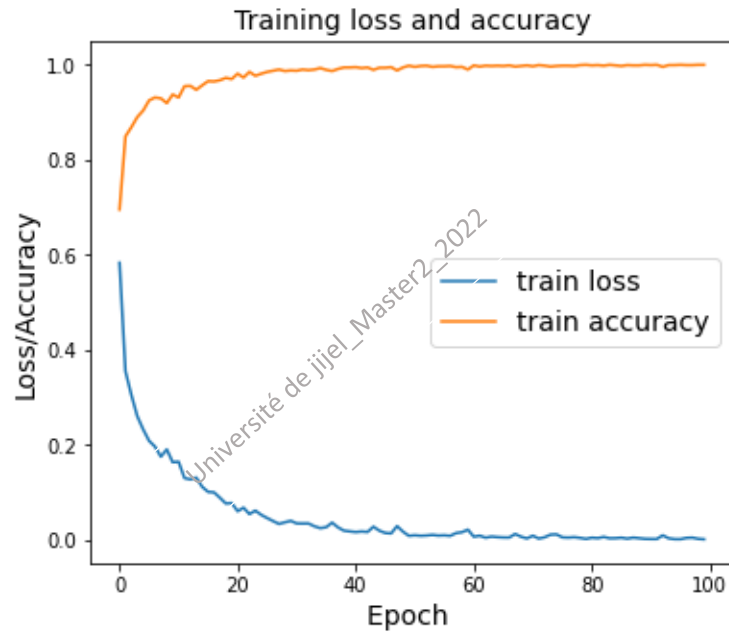


Figure 4.2. La fonction Coste (loss) et le taux de l'apprentissage pour la détection.

Après l'apprentissage et l'évaluation du modèle, nous utilisons le 20% des données réservées pour tester la précision et les pertes de la validation, les résultats sont montrés dans la figure 4.3 qui suit :



Figure 4.3. La fonction cout (loss) et le taux de la validation pour la détection.

Pour la précision de la validation nous avons obtenu 98% ; c'est-à-dire notre système peut classer 98% des images dans leurs réelles.

4.4.1.2. Matrice de confusion

Le tableau suivant montre les résultats cités précédemment (la précision, rappel et f1-score).

Table 4.1. Précision, rappel et f1-score de la détection.

	précision	rappel	f1-score	support
Classe 0	0.97	0.99	0.98	200
Classe 1	0.99	0.96	0.98	200
Taux			0.98	400
macro avg	0.98	0.98	0.98	400
weighted avg	0.98	0.98	0.98	400

La matrice de confusion obtenue est présentée dans la figure 4.4, on observe que 198 images qui sont réellement fautives ont été mises dans la classe fautive (TP), et 2 images qui ne sont pas fautives mais elle aussi mises dans cette classe (FN). En outre, 7 images sont fautives en réalité mais mises dans la classe normale, et 193 images normale sont également mises dans cette classe.

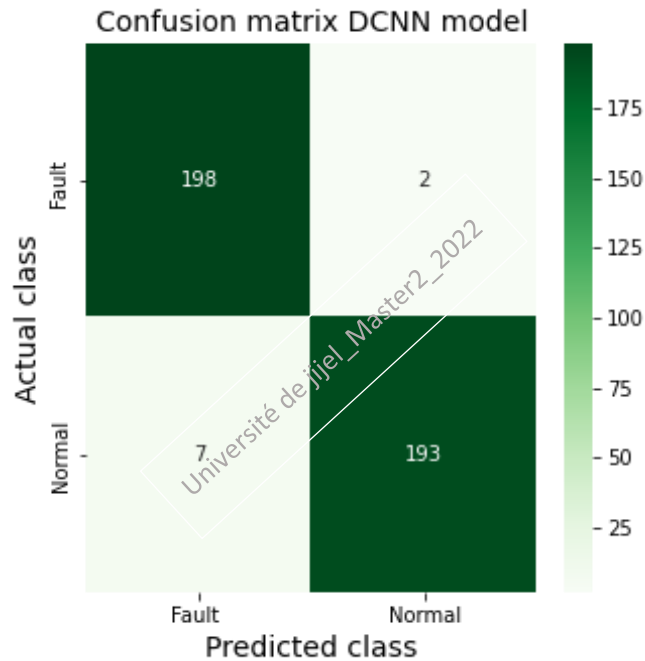


Figure 4.4. La matrice de confusion de la détection.

4.4.1.3. Courbe ROC

La figure 4.5 suivante (a) montre la courbe ROC obtenue, et (b) est un zoom sur la zone gauche pour bien observer les résultats. On a en abscisse le taux de faux positifs et en

ordonnée le taux de vrais positifs. On constate que AUC (Area Under Curve) est égale à 1 et en observant le seuil donné par la courbe (a) on peut dire que les résultats sont parfaits.

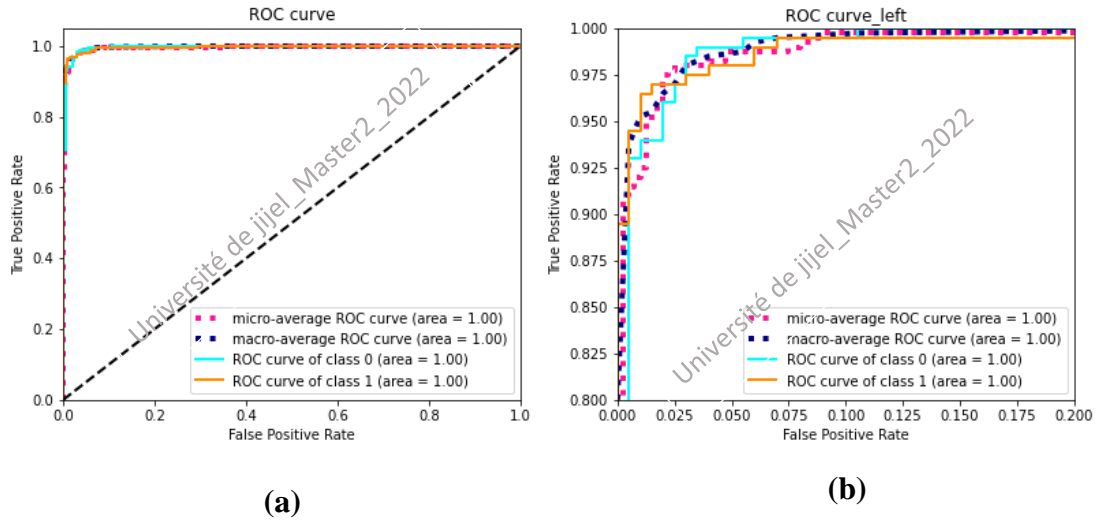


Figure 4.5. La courbe ROC pour la détection.

4.4.2. La classification

4.4.2.1. Précision d'apprentissage et validation

Pour la multi classification ; nous avons obtenu 100% comme précision d'apprentissage, et 0.02 comme perte, ces résultats sont montrés dans la figure ci-dessous :

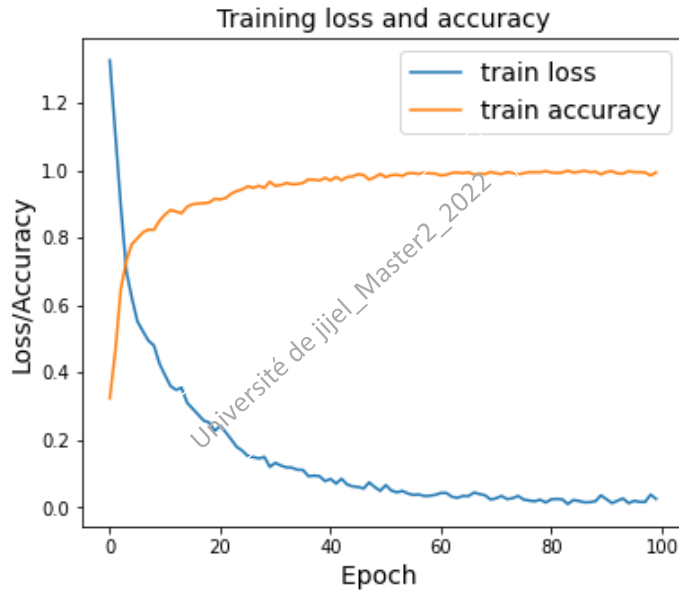


Figure 4.6. La fonction cout (loss) et le taux de l'apprentissage pour la classification.

En utilisant 67% pour l'apprentissage ; le 33% qui reste est utilisé pour faire le test ; les résultats sont présentés dans la figure 4.7. Nous avons obtenu 0,91 comme précision de la validation et 0,54 comme perte.

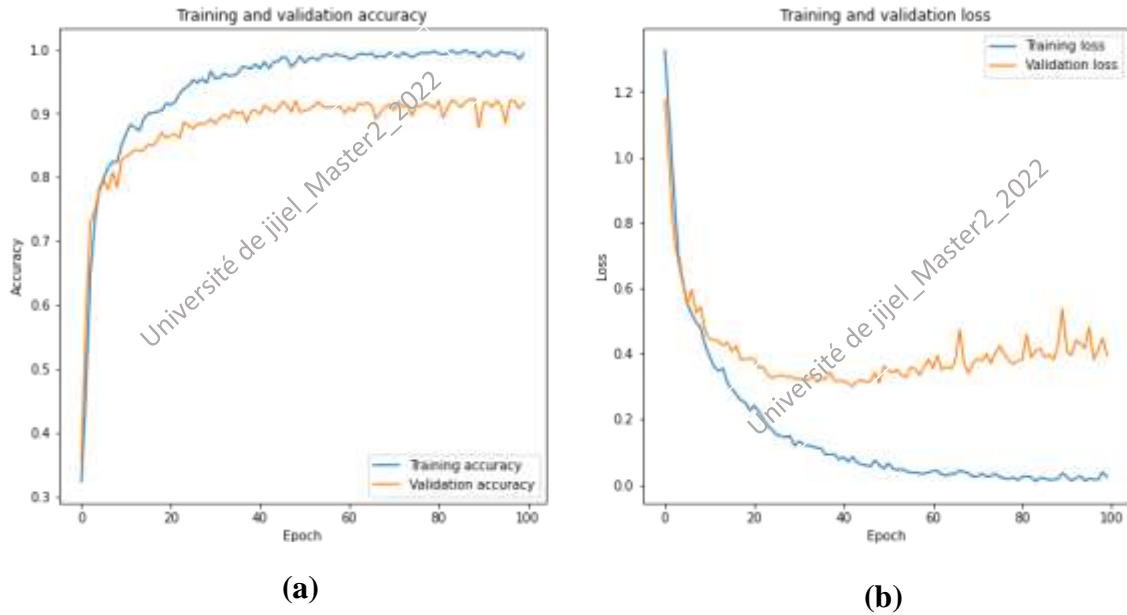


Figure 4.7. Les pertes et le taux de la validation pour la classification.

4.4.2.2. Matrice de confusion

Les résultats détaillés sont montrés dans le tableau 4.2 qui suit :

Table 4.2. Précision, rappel et f1-score de la classification.

	précision	rappel	f1-score	support
Classe 0 (F1)	0.87	0.91	0.89	165
Classe 1 (F2)	0.95	0.95	0.95	166
Classe 2 (F3)	0.92	0.92	0.92	165
Classe 3 (F4)	0.95	0.90	0.92	167
Taux			0.91	663
macro avg	0.92	0.92	0.92	663
weighted avg	0.92	0.92	0.92	663

En observant la figure 4.8 qui présente la matrice de confusion obtenue, le diagonal montre les images mises dans leurs réelles classes. Par contre les autres cellules montrent les images qui sont réellement d'une classe et mises dans une autre, par exemple la cellule 2 (1^{ère} ligne et 2^{ème} colonne) montre les 5 images de la classe (F1) mais mises dans (F2).

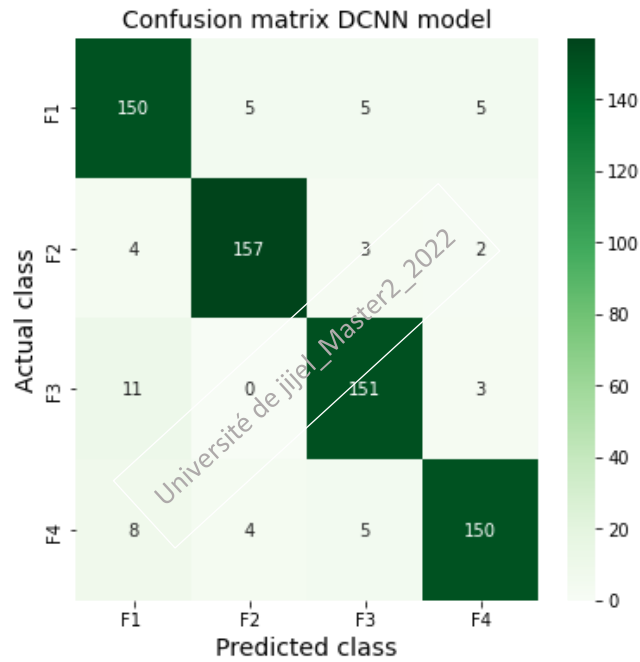


Figure 4.8. La matrice de confusion de la classification.

4.4.2.3. Courbe ROC

La figure 4.9 suivante (a) montre la courbe ROC obtenue, et (b) est un zoom sur la zone gauche pour bien observer les résultats. On a en abscisse le taux de faux positifs et en ordonnée le taux de vrais positifs. On constate que AUC (Area Under Curve) est égale à 0.99 pour les deux classes (1, 2) et 0.98 pour les deux classes (0, 3). En observant le seuil donné par la courbe (0.98) on peut dire que les résultats sont aussi bien mais ne sont pas parfaits.

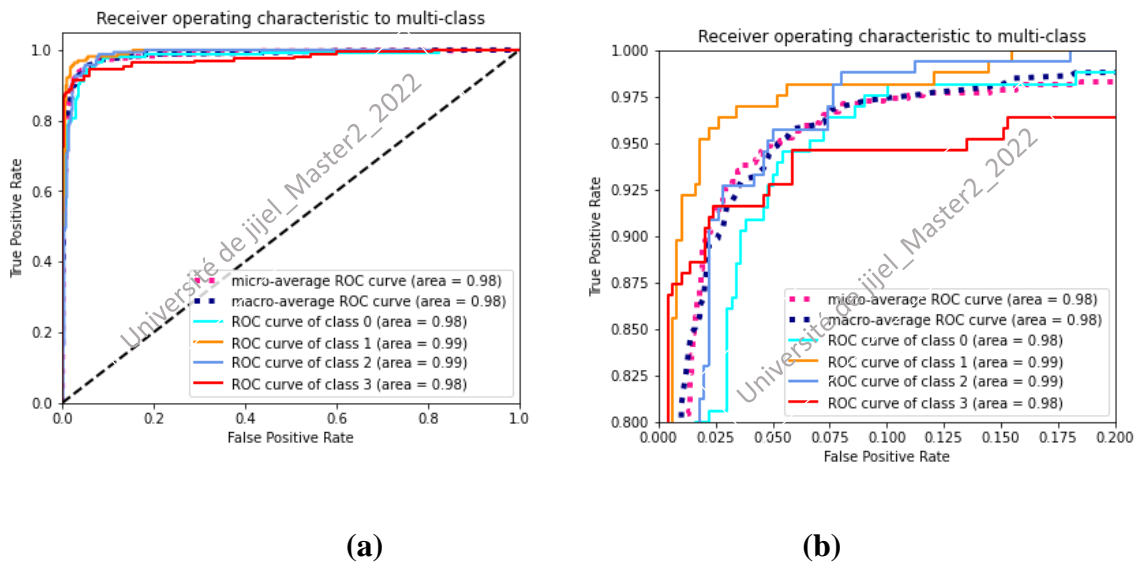


Figure 4.9. La courbe ROC-AUC de la classification.

4.5. Résultats du test et simulation

4.5.1. Sauvegarde et optimisation des modèles

Afin de faire utiliser les deux modèles ; nous avons premièrement les sauvegardé sous le format h5, puis les optimiser avec le convertisseur tflite (tensorflow lite) afin de les utiliser sur le Raspberry.

4.5.2. Test du programme principal

Avant tout, nous avons testé notre programme principal pour confirmer l'envoi des messages, en affichant la classe prédise sur le terminal. Les résultats sont montrés dans la figure 4.10 qui suit ; la figure (a) présente les résultats de la prédiction d'une image quelleconque, et la figure (b) montre le câblage du GSM avec le Raspberry l'envoi des messages en cas d'un défaut du dépôt de la poussière sur le module PV.



(a)

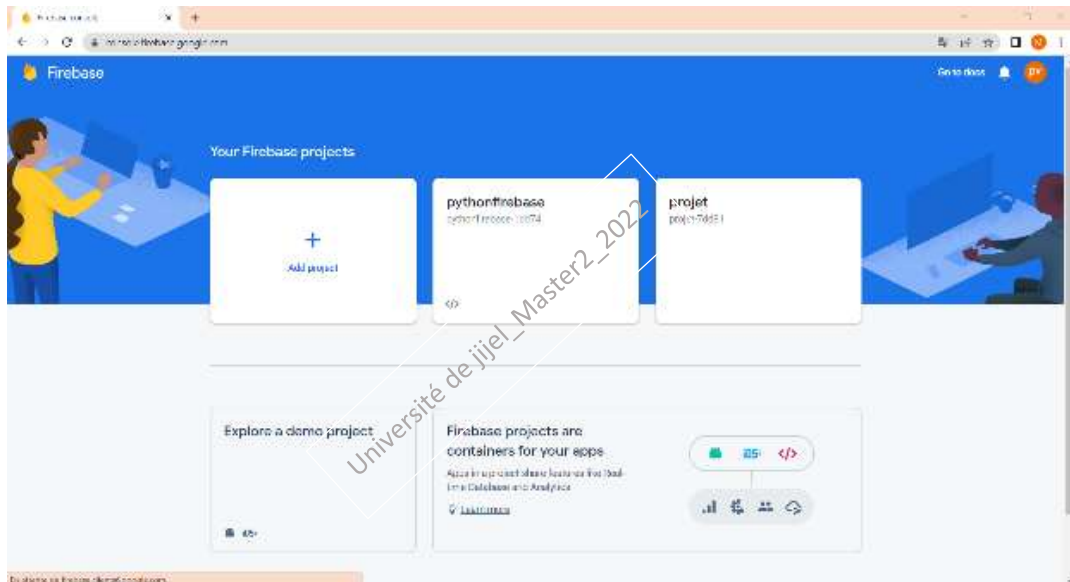


(b)

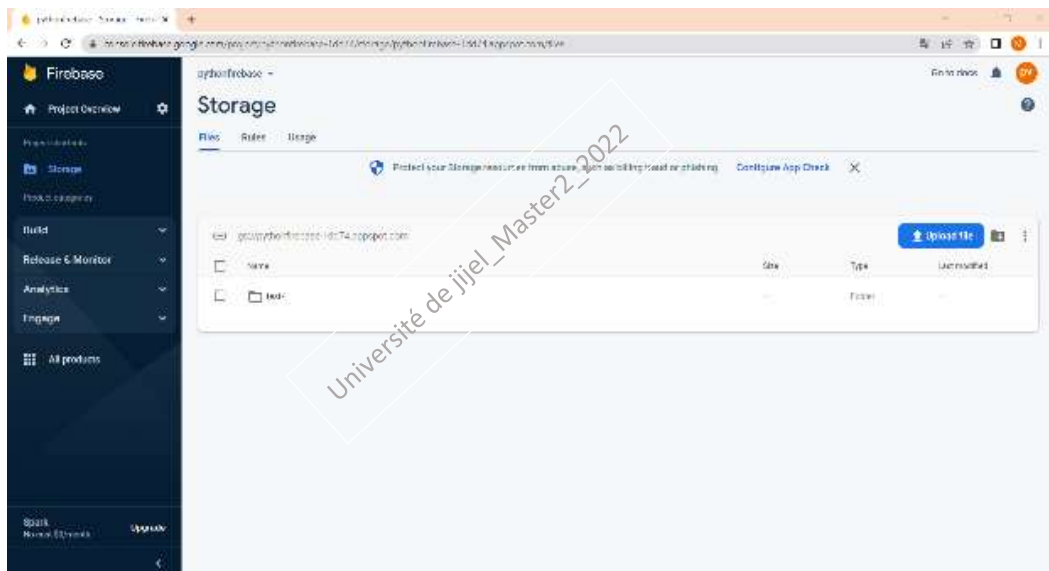
Figure 4.10. Réalisation et vérification d'envoi des messages.

4.5.3. Stockage des données

Pour stocker les images nous avons utilisé Firebase Storage comme un site de stockage des données, puis accéder aux images avec un code python et la Key.json, la page du Firebase est montrée dans la figure 4.11, la figure 4.11.a est la page de console, et 4.11.b est la page du storage créé dans une application du projet.



(a)



(b)

Figure 4.11. Firebase.

4.6. Utilisation de l'interface graphique

Notre système est mis sous application comme présenté dans la figure 4.12.

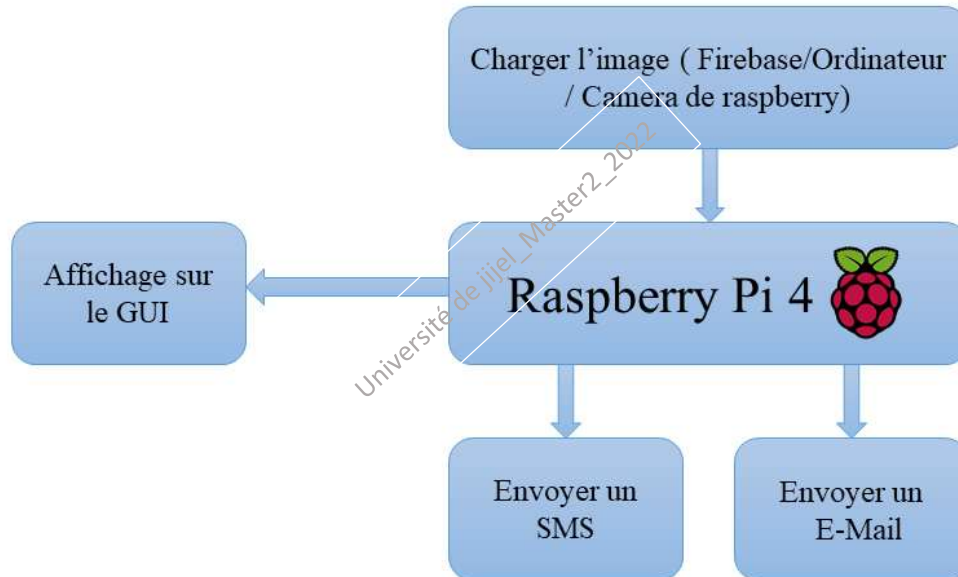


Figure 4.12. Fonctionnement du système.

L'interface graphique que nous avons créée est composée de deux pages ; la première page est celle de l'accueil (Start page), la deuxième contient les onglets de l'affichage. Ces deux pages sont montrés dans la figure 4.13 respectivement.



Figure 4.13. L'interface graphique développée.

En choisissant la source de l'image, et la chargeant dans le programme, nous faisons la détection et la classification s'il y'a un défaut selon l'organigramme illustré dans le chapitre 3, nous obtenons les résultats présentés dans la Figure 4.14.

Cette interface (Software) est donnée à l'utilisateur accompagnant avec le Raspberry comme une application en format .exe pour faire l'affichage.



(a)

(b)



(c)



(d)



(e)



(f)



(g)

Figure 4.14. Affichage sur l'interface graphique.

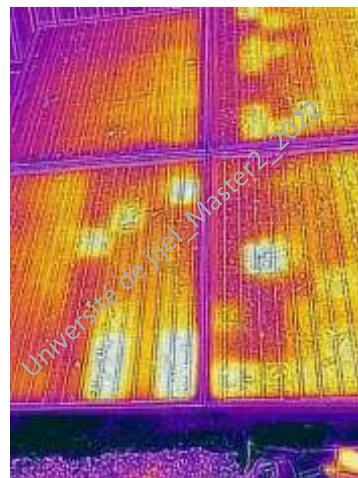
Dans les figures 4.14 (a, b, d, e, f) ; pour l'image (b) il n'y a pas un défaut, et le système a bien classé cette image dans la classe normale, pour le reste, il détecte qu'il y'a un défaut et aussi bien classé ce défaut.

Cependant pour l'image (c) ; il y's un défaut mais le système ne le détecte pas, car la précision de la détection n'est pas 100%. Pour l'image (g), il s'agit du défaut de la diode endommagée mais il le mit dans la classe du module PV court-circuité.

De plus, nous avons prend de nouvelles images avec une autre caméra thermique associées avec le téléphone portable (voir figure 4.15), et le test que nous avons fait été sur les images prises par la nouvelle caméra.



(a)



(b)



(c)

Figure 4.15. Test d'une nouvelle image.

Dans ce cas, le système détecte le défaut mais il ne l'a pas bien classifié, c'est dû à la qualité d'image car elle est prise par une caméra thermique non professionnelle avec une résolution très faible ; en outre l'image est brute (n'a pas été traitée).

4.7. Comparaison avec d'autres méthodes

En outre, nous avons testé d'autres approches pour faire la classification. Premièrement ; une méthode connue d'apprentissage automatique le K-PPV(KNN), et le réseau célèbre ResNet50 de l'apprentissage profond. Les résultats comparatifs sont montrés dans le tableau 4.1 suivant :

Tableau 4.3. Résultats de la comparaison.

Méthode	Précision %
K-PPV	70
ResNet50	100
Notre Modèle	92

Pour le K-PPV ; nous avons trouvé que le meilleur nombre pour les voisins est K=15.

4.8. Conclusion

Dans ce chapitre, nous avons présenté les résultats obtenus au cours de la réalisation du projet. En montrant la précision de nos modèles et donc la précision du système généralement.

Nous avons également présenté les résultats du test fournis par les modèles dans la phase du test/validation. Terminant par l'application du deux classificateurs sur des nouvelles images et l'affichage sur une interface graphique.



Conclusion générale



Conclusion générale

Ce mémoire avait pour ambition de réaliser un système semi-automatique détectant et classifiant les causes provoquant le problème des points chauds dans un générateur photovoltaïque. Deux algorithmes basés sur un DCNN ont été appliqués pour développer les classificateurs (détection et identification des défauts). Les résultats de la simulation ont montré une précision tout à fait acceptable pour les deux classificateurs (détection 98 % et classification 92 %).

Les modèles optimisés sont ensuite implémentés dans un Raspberry PI 4 pour une application en temps réel. Les essais expérimentaux indiquent la capacité d'un tel système embarqué à détecter une défaillance avec une précision acceptable, tandis que la classification multi-classes des défauts doit être améliorée.

En commençant dans le premier chapitre par une brève présentation des générateurs photovoltaïques, abordant ensuite; des défauts qui peuvent être provoqués dans ces générateurs. En terminant finalement ; par les méthodes de diagnostic de ces défauts.

Le deuxième chapitre, est mis afin de présenter brièvement le traitement des images, et les méthodes de leur classification, en passant de l'apprentissage automatique à l'apprentissage profond, présentant les différents types de ce dernier, et terminant par la présentation des réseaux de neurones convolutifs.

Nous avons mis en œuvre une méthode de diagnostic des générateurs photovoltaïques dans le troisième chapitre. Ce dernier encadre les étapes de notre travail, tout en présentant des échantillons de la base de données utilisée et l'architecture des classificateurs.

Terminant, enfin par le quatrième chapitre, qu'est une présentation détaillée des résultats obtenus. Nous avons abordé dans ce chapitre, les résultats de la phase d'apprentissage, puis ceux du test et validation. La sauvegarde et l'optimisation des modèles, et les résultats d'application expérimentale ont été présentés dans le même chapitre.

C'était prévu que le système réalisé fonctionne en temps réel, en accédant directement aux images prises par une caméra associée à un drone survolant le système PV, mais à cause du manque de matériels et la courte période allouée à ce travail, nous avons juste sauvegardé les images (PC, Firebase) et puis les utilisés.

Ce travail a été effectué au sein du laboratoire des énergies renouvelables, et en collaboration avec le centre de recherche UDES.

Comme continuité de ce modeste travail et afin d'améliorer les résultats de classification, nous proposons d'utiliser une base de données large.

Comme perspectives, nous allons :

- Améliorer les résultats de la précision obtenus.
- Faire la détection, localisation et classification.
- Développer l'application créée.

Références

- [1]. [https://energyeducation.ca/encyclopedia/Photovoltaic_system#:~:text=A%20photovoltaic%20\(PV\)%20system%20is,massive%20utility%2Dscale%20generation%20plants](https://energyeducation.ca/encyclopedia/Photovoltaic_system#:~:text=A%20photovoltaic%20(PV)%20system%20is,massive%20utility%2Dscale%20generation%20plants).
[Consulté le 10 Juin 2022].
- [2]. Mellit, A., Tina, G.M. and Kalogirou, S.A., 2018. "Fault detection and diagnosis methods for photovoltaic systems: A review". *Renewable and Sustainable Energy Reviews*, 91, pp.1-17.
- [3]. Su, Y., Tao, F., Jin, J. and Zhang, C., 2021. "Automated overheated region object detection of a photovoltaic module with thermography image". *IEEE Journal of Photovoltaics*, 11(2), (pp.535-544).
- [4]. Li, X., Yang, Q., Lou, Z. and Yan, W., 2018. "Deep learning-based module defect analysis for large-scale photovoltaic farms". *IEEE Transactions on Energy Conversion*, 34(1), (pp.520-529).
- [5]. Li, X., Li, W., Yang, Q., Yan, W. and Zomaya, A.Y., 2019. "An unmanned inspection system for multiple defects detection in photovoltaic plants". *IEEE Journal of Photovoltaics*, 10(2), (pp.568-576).
- [6]. Korkmaz, D. et Acikgoz, H., 2022. An efficient fault classification method in solar photovoltaic modules using transfer learning and a multi-scale convolutional neural network. *Engineering Applications of Artificial Intelligence*, 113, (p.104959).
- [7]. <https://www.photovoltaïque.info/fr/info-ou-intox/lenergie-solaire/de-la-lumiere-a-lelectricite/>. [Consulté le 28 Juin 2022].
- [8]. Djalab, A. et Bouafia, L., "Diagnosis of faults in a renewable energy system based on artificial intelligence techniques : Application on a photovoltaic system," Thèse de Doctorat en Automatique : Diagnostiques Industriels, Université Ziane Achour, Djelfa, Algérie ,2020.
- [9]. Long, B., "Détection et localisation de défauts dans un système photovoltaïque," Thèse de Doctorat en Energie électrique, Université de Grenoble, France, 2011.
- [10]. Benarroudj, W., "Détection et compensation de défaut dans un système photovoltaïque," Mémoire de Master en Automatique : Automatique et Informatique industriel, Université de Mohamed El Bachir El Ibrahimi, B.B.A,2021.

- [11]. Mellit, A., 2020. "Recent applications of artificial intelligence in fault diagnosis of photovoltaic systems. In A. Mellit and M. Benghanem, (Editors). *A Practical Guide for Advanced Methods in Solar Photovoltaic Systems*" (pp. 257-271). Springer, Cham..
- [12]. Zairi, S., Boubiada, S., "Etude et Dimensionnement d'un centrale Photovoltaïque," Mémoire de Master en Electrotechnique : Energies Renouvelables, Université de Mohamed Khider, Biskra ,2020.
- [13]. Aouchiche, N. 2018."Défauts liés aux systèmes photovoltaïques autonomes et techniques de diagnostic-Etat de l'art." *Journal of Renewable Energies* 21.2 : 247-265.
- [14]. Khakadja, M., et Benierbah, S., "Traitement d'image," Cours de Master en Systèmes des Télécommunications, Université des frères Mentouri, Canstantine.
- [15]. Mezzoug K, "Traitement et analyse des images numériques," Cours de Master en Génie Informatique, Université Ibn Khaldoun, Tiaret,2019-2020
- [16]. <https://www.techno-science.net/glossaire-definition/Histogramme-imagerie-numerique.html> . [Consulté le 16 Juin 2022].
- [17]. [https://www.delftstack.com/fr/howto/matlab/matlab-image-histogram/#:~:text=Nous%20pouvons%20utiliser%20la%20fonction,de%20la%20fonction%20imhist\(\)%20](https://www.delftstack.com/fr/howto/matlab/matlab-image-histogram/#:~:text=Nous%20pouvons%20utiliser%20la%20fonction,de%20la%20fonction%20imhist()%20). [Consulté le 16 Juin 2022].
- [18]. Boumimez, A., et Djidjeli, M., "Système de reconnaissance automatique des plaques d'immatriculation en utilisant l'apprentissage artificiel, "Mémoire de Master en Electronique des Systèmes Embarqués, Université Mohamed Seddik Benyahia, Jijel,2021.
- [19]. Zerougui, A., " Traitement d'images monochromes : Détection de contours, Filtrage (Spatial et et Segmentation par Réseaux de Neurons, "Mémoir de Master en Informatique Industrielle,Université Larbi Ben M'hidi, Oum El Bouaghi,2017.
- [20]. Bendahmane, D., "Une classification robuste d'images, " Mémoire de Master en informatique- IVA, Université Mohamed Khider, Biskra,2019.
- [21]. Ghabri, F, Z., et Ben Namia, H., "La logique floue pour Classification Des Feuilles de vigne," Mémoire de Master en informatique-informatique industriel, Université KASDI MARBAH, Ouargla,2016.
- [22]. K, S., Zeeshan, K., and Anurag J.,2012. "A review of content-based image classification using machine learning approach." *International Journal of Advanced Computer Research* 2.3: 55.
- [23]. Horning, N. 2010. "Random Forests: An algorithm for image classification and generation of continuous fields data sets." *Proceedings of the International Conference*

- on Geoinformatics for Spatial Infrastructure Development in Earth and Allied Sciences, Osaka, Japan. Vol. 911.*
- [24]. Merdoud, K., et Bousbaine, K., “*Détection de maladies par traitement d'image,*” Mémoire de Master en Informatique – Génie Système Informatique, Université Akli Moand Oulhadje, Bouira, 2018.
- [25]. Lefza, A., “*Application des réseaux de neurones profonds pour la prédiction de la puissance produite par un micro-réseau, dédié au chargement des véhicules électriques,*” Mémoire de Master en Electronique – Electronique des Systèmes Embarqués, Université Mohammed Seddik BenYahia, Jijel, 2020.
- [26]. Alzubaidi, L., et al., 2021. "Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions." *Journal of Big Data* 8.1: 1-74.
- [27]. <https://www.natural-solutions.eu/blog/histoire-du-deep-learning> . [Consulté le 22 juin 2022].
- [28]. Idoui, R., Himeur, A., “*Reconnaissance d’images par les réseaux de neurones convolutionnels (CNN),*” Mémoire de Master en Electronique – Electronique des Systèmes Embarqués, Université Mohammed Seddik BenYahia, Jijel, 2020.
- [29]. Gu, Hao, et al. 2019. "Blind channel identification aided generalized automatic modulation recognition based on deep learning." *IEEE Access* 7: 110722-110729. [blind channel]
- [30]. Moganam, Praveen, K., and Denis A, S, S., 2022. "Deep learning and machine learning neural network approach for multi-class leather texture defect classification and segmentation." *Journal of Leather Science and Engineering* 4.1 (1-21).
- [31]. Gholamalinezhad, Hossein, and Khosravi, H. 2020, "Pooling methods in deep neural networks, a review." *arXiv preprint arXiv:2009.07485*.
- [32]. Mellit A., and Soteris K., 2022. Machine learning and deep learning methods for fault diagnosis of photovoltaic systems. In: Mellit A., and Kalogirou S. (Editors), *Handbook of Artificial Intelligence Techniques in Photovoltaic Systems: Modeling, Control, Optimization, Forecasting and Fault Diagnosis, Academic Press, ISBN: 9780128206416, 1st Ed, Elsevier.*
- [33]. <https://ledatascientist.com/google-colab-le-guide-ultime/>. [Consulté le 24 Juin 2022].
- [34]. Mellit, A., and Soteris K., 2021. "Artificial intelligence and internet of things to improve the efficacy of diagnosis and remote sensing of solar photovoltaic systems:

Challenges, recommendations, and future directions." *Renewable and Sustainable Energy Reviews* 143: 110889.

- [35]. <https://kobia.fr/classification-metrics-matrice-de-confusion/>. [Consulté le 27 Juin 2022].
- [36]. <https://www.analyticsvidhya.com/blog/2020/06/auc-roc-curve-machine-learning/#:~:text=The%20Area%20Under%20the%20Curve,the%20positive%20and%20negative%20classes>. [Consulté le 1 Juillet 2022].