

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche
Scientifique



L'Université Echahid Hamma Lakhdar d'El Oued

Faculté de Technologie

Mémoire de Fin d'Étude

En vue de l'obtention du diplôme de

MASTER ACADEMIQUE

Domaine : Sciences et Technologie

Filière : Electrotechnique

Spécialité : Réseaux Electrique

Thème

**Nouvelle Méthode de contrôle
d' isolateur de haute tension**

Présenté par :



MEHELLOU Abderrahmane
DEBBAR Khalifa

Encadré par :

Dr. GUIA Talal

Soutenu en 2021

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ



REMERCIEMENTS

C'est avec un réel plaisir que nous réservons ces lignes en signe de gratitude et de profonde reconnaissance à tous ceux qui, de près ou de loin, ont contribué à la réalisation et à l'aboutissement de ce travail.

Avant tout, nous remercions Dieu tout puissant pour m'avoir aidé à réaliser ce modeste travail.

Nous tiens tout d'abord à remercier : Dr. GUIA Talal qui est eu l'amabilité de nous encadrer pour la réalisation de ce travail Et qui dirigé ce travail depuis les premiers instants .

Je remercie de même tous les enseignants de notre département qui ont énormément contribué à nous formation .

Enfin, je tiens également à remercier toutes les personnes qui ont participé de près ou de loin à la réalisation de ce travail.

Dédicace

Je dédie cet humble travail

A celui qui m'a ouvert les portes et m'a donné tendresse et courage

Pour ceux qui sont tristes pour me faire plaisir

Pour ceux qui attendent ce jour avec impatience :

"Chère maman"

Pour ceux qui mettent tant d'efforts dans mon bonheur

Qui a rêvé de voir ce jour

Qui m'a guidé et m'a pris les secrets de la vie

" mon père "

"à mes frères"

A toutes les tantes, oncles, oncles et tantes dans la vie éternelle aussi

A tous mes amis, mes merveilleux frères et soeurs

A tous ceux qui m'ont aidé de près ou de loin.

pour tous les proches

abderrahmane

khalifa



Résumé

La cause principale de l'apparition du phénomène de contournement est l'accumulation des agents polluantes sur la surface d'isolateur de haute tension, ce qui conduit à coupures d'alimentation électrique, il est important de connaître l'état des isolateurs pour éviter les coupures et augmenter la fiabilité du réseau. Ce travail fournit une solution pour surveiller l'état de surface d'un isolateur de haute tension par l'utilisation des techniques d'intelligence artificielle tel que coco model et nous avons intégré un algorithme de reconnaissance sur une carte de Raspberry pi qui nous permet de savoir l'état de surface d'isolateur qui installer sur les pylônes extérieurs.

Mots clés: Pollution- Isolateurs de haute tension - Surveillance– Techniques d'intelligence artificielle- model coco- Raspberry pi .

الملخص

السبب الرئيسي لحدوث ظاهرة الالتفاف هو تراكم الملوثات على سطح عازل الجهد العالي ، مما يؤدي إلى انقطاع التيار الكهربائي ، من المهم معرفة حالة العوازل لتجنب انقطاع التيار وزيادة موثوقية الشبكة. يوفر هذا العمل حلاً لمراقبة حالة سطح عازل الجهد العالي من خلال استخدام تقنيات الذكاء الاصطناعي مثل نموذج كوكو وقمنا بدمج خوارزمية التعرف على لوحة رسبري باي التي تتيح لنا معرفة حالة سطح العازل يتم تثبيتها على الأبراج الخارجية.

الكلمات المفتاحية: التلوث - عوازل الجهد العالي - المراقبة - تقنيات الذكاء الاصطناعي - نموذج كوكو – رسبري باي .

ABSTRACT

The main cause of the occurrence of the bypass phenomenon is the accumulation of pollutants on the surface of high voltage insulator, which leads to power cuts, it is important to know the condition of the insulators to avoid outages and increase network reliability. This work provides a solution to monitor the surface condition of a high voltage insulator through the use of artificial intelligence techniques such as coco model and we have integrated a recognition algorithm on a Raspberry Pi board that allows us to know the surface condition of the insulator to be installed on the exterior pylons.

Keywords: Pollution - High voltage insulator – Surveillance - Artificial intelligence techniques -model coco- Raspberry pi .

Liste des figures

Chapitre I

Figure I.1. Les Caractéristiques d'un isolateur.....	5
Figure I.2. Isolateur rigide	7
Figure I.3. Isolateur long fut.	7
Figure I.4. Isolateurs à capot et tige.....	8
Figure I.5. Isolateur capot et tige avec profil standard.....	9
Figure I.6. Profil antibrouillard (forme A).....	9
Figure I.7. Profil antibrouillard (forme B).....	9
Figure I.8. Profil sphérique.....	10
Figure I.9. Profil plat.....	10
Figure I.10. Isolateur composite.....	11
Figure I.11. La prospérité hydrophobe	11
Figure I.12. Vieillissement d'un isolateur composite.....	11

Chapitre II

Figure II.1 : le schéma de montage virtuel pour visualiser un signal sur LabVIEW.....	22
Figure II.2. Afficheur de l'oscilloscope virtuel sur LabVIEW.....	23
Figure II.3 : Dispositif d'essais et de mesure.....	23
Figure II.4 : Chaîne d'isolateurs U120B étudiée.....	24
Figure II.5 : Surfaces interne et externe d'un élément de la chaîne d'isolateur.....	25
Figure II.6 : Représentation du spectre d'amplitude.....	26
Figure II.7. Représentation du signal du courant de fuite correspondant à une faible pollution.....	27
Figure II.8. Représentation du signal du courant de fuite correspondant à une moyenne pollution.....	27
Figure II.9. Représentation du signal du courant de fuite correspondant à une forte pollution...	28

Figure II.10. Exemple de détection d'objets.....	28
Figure II.11. Illustration de la différence entre la classification et la détection d'objets....	29
Figure II.12. Exemple de génération de graphe de calcul à partir de code (Tensorflow)...	31

Chapitre III

Figure III.1. Création de l'environnement de travail.....	35
Figure III.2. (A) (B) (C) List de Package	38
Figure III.3. Création des fichier name_pb2.....	40
Figure III.4. Installation des packages Jupiter notebook pour la détection des objets	41
Figure III.5. Teste de l'installation des packages Jupiter notebook	42
Figure III.6. Assemblage des images	43
Figure III.7. Isolateur propre.....	44
Figure III.8. Isolateur pollue.....	44
Figure III.9. Enregistre un fichier .xml.....	45
Figure III.10. Création des fichier csv	45
Figure III.11. Les fichier csv créés	46
Figure III.12. Génération des numérotations aux objets	46
Figure III.13. Créer des fichiers TF Record de train.....	47
Figure III.14. Créer des fichiers de TF Record de test	47
Figure III.15. Assurez-vous créer TF Record de test et train	47
Figure III.16. Fichier labelmap.pbtxt	48
Figure III.17. Classification des fichiers	48
Figure III.18. Classeur train_input_reader.....	48
Figure III.19. Classeur eval_input_reader	49
Figure III.20. Exécution de l'entraînement	49
Figure III.21. Contrôler l'évaluation d'entraînement	50

Figure III.22. (A)(B)(C): Les courbes de losses.....	51
Figure III.23. Dossiers d'entraînement	51
Figure III.24. Création du graphe d'inférence	52
Figure III.25. Crée un fichier Frozen_inference_graph.pb.....	52
Figure III.26. Ouverture de l'interface python	53
Figure III.27. Configuration du modèle de détection de l'état de surface des isolateurs.....	53
Figure III.28. Isolateur propre	54
Figure III.29. Isolateur pollutes	54
Figure III.30. Chaîne isolateurs propres	54
Figure III.31. Chaîne isolateurs pollués.....	55

Chapitre IV

Figure IV.1. Raspberry Pi 3 model b.....	56
Figure IV.2. Camera USB.....	57
Figure IV.3. microSD card 32GB.....	58
Figure IV.4. Alimentation de Raspberry pi.....	58
Figure IV.5. câble HDMI.....	58
Figure IV.6. Clavier et souris.....	59
Figure IV.7. Ventilateur.....	59
Figure IV.8. L'insigne de opencv	60
Figure IV.9. Le site officiel pour l'installation du système d'exploitation sur la carte mémoire.....	61
Figure IV.10. Premier fonctionnement de raspberry pi.....	61
Figure IV.11. Création des fichiers fondamentaux	63
Figure IV.12. Terminal <i>IDE</i> du Python.....	63
Figure IV.13. Premier code.....	64

Figure IV.14. Exécution du code et l'enregistrement des photos pour les isoleurs	64
Figure IV.15. Deuxième code.....	65
Figure IV.16. Analyse des données d'images.....	65
Figure IV.17. Détection d'objet par l'utilisation de Raspberry Pi	66

Liste des tableaux

Chapitre I

Tableau I.1. Classification de la pollution selon le type d'environnement.....	17
---	----

Chapitre III

Tableau III.1. Modules et toolbox nécessaires.....	35
---	----

Tableau III.2. Code d'installation des packages.....	36
---	----

Tableau III.3. classe de image	43
---	----

Chapitre IV

Tableau. IV.1. Installons OpenCV, face_recognition et imutils.....	62
---	----

Listes des abréviations et des symboles

L_r : La ligne de fuite.

CEI: Commission électrotechnique internationale

R_s : résistance surfacique

R_v : résistance volumique

SIR: Silicon Rubber.

Sommaire

INTRODUCTION GENERALE.....	1
Chapitre I : Isolateurs de haute tension et phénomène de pollution	
I.1. Introduction.....	3
I.2. Les isolateurs	3
I.2.1. Définition des isolateurs de haute tension	3
I.2.2. Fonctionnement et constitution d'un isolateur	4
I.3. Quelques définitions concernant les isolateurs.....	4
I.3.1. Courant de fuite critique.....	4
I.3.2. Tension de tenue (Tension d'essai).....	4
I.3.3. Contournement.....	4
I.3.4. Contrainte de Contournement.....	5
I.3.5. Tension de Contournement.....	5
I.3.6. Courant de fuite.....	5
I.3.7. Rigidité diélectrique.....	6
I.3.8. Résistivité diélectrique.....	6
I.3.9. Conductivité électrique.....	6
I.4. Types d'isolateurs	6
I.4.1. Isolateurs rigides	6
I.4.2. Isolateurs suspendus ou éléments de chaîne	7
I.4.3. Isolateur longfut	7
I.4.4. Isolateur capot et tige	8
I.4.4.1. Les différents profils d'Isolateur capot et tige	8
I.4.4.2. Profil standard	8

I.4.4.3. Profil anti brouillard (forme A)	9
I.4.4.4. Profil antibrouillard (forme B)	9
I.4.4.5. Profil sphérique	10
I.4.4.6. Profil plat	10
I.5.Choix des isolateurs	10
I.6.Matériaux isolants utilisés pour la fabrication des isolateurs	11
I.6.1. Céramique	11
I.6.2. Le verre	11
I.6.3. Matériaux synthétiques	12
I.7. Défauts des isolateurs	12
I.7.1. Défauts électriques	13
I.7.2. Défauts mécaniques	13
I.7.3. Défauts dus à la manipulation	13
I.8. Vieillissement des matériaux isolants	14
I.8.1. Farinage	14
I.8.2. Faïençage	14
I.8.3. Craquelures	14
I.8.4. Erosion	14
I.8.5. Acheminement	14
I.9.Pollution des isolateurs	14
I.9.1. Sources de pollution	15
I.9.1.1 Types de pollution	15
I.9.1.1.1 Pollution naturelle	15
I.9.1.1.1.1. Pollution marine	16
I.9.1.1.1.2. Pollution désertique	16

I.9.1.1.2. Pollution industrielle	16
I.9.1.1.3. Pollution mixte	16
I.9.1.2. Classifications de types de pollution	16
I.9.2. Formation et répartition des couches polluantes	17
I.9.3. Conséquences de la pollution	18
I.10. Conclusion	19
 Chapitre II : Méthodes de diagnostic et de détection des objets	
II.1. Introduction	20
II.2. Les méthodes de diagnostic	20
II.2.1. Définition	20
II.2.2. Prédiction du degré de sévérité de pollution	20
II.2.3. Enregistrement du signal du courant de fuite	21
II.2.4. Origines de LabVIEW	21
II.2.5. Utilité de LabVIEW	22
II.2.6. Initiation aux instruments virtuels de LabVIEW	22
II.3. Isolateurs testés	23
II.4 Procédures de test	24
II.4.1. Préparation des chaines d'isolateurs	24
II.4.2. Préparation des solutions polluantes	24
II.4.3. Méthode d'application de la pollution	24
II.5 Circuits de mesure	25
II.5.1. Mesure de la tension de contournement	25
II.5.2. Mesure du courant de fuite	25
II.6 Exploitation du signal du courant de fuite pour la prédiction de la sévérité de Pollution ..	25
II.6.1 L'analyse spectrale	26

II.6.2. Application de l'analyse spectrale aux signaux du courant de fuite	26
II.7. Méthodes de détection des objets	28
II.7.1. Définition	28
II.7.2. OpenCV	29
II.7.2.1. Définition OpenCV	29
II.7.2.2. Fonctionnalités	29
II.7.2.2.1. Traitement d'images	29
II.7.2.2.2. Traitement vidéo	29
II.7.2.3. Modules OpenCV	30
II.7.3. Tensorflow	31
II.7.3.1. Définition	31
II.7.3.2. Fonctionnement	31
II.7.3.3. Les composants	32
II.8. Conclusion	33

Chapitre III : Détection de l'état de surface d'isolateur par les techniques de TensorFlow

III.1. Introduction.....	34
III.2. Abstraite	34
III.3. Modules et toolbox nécessaires	34
III.4. Processus de création d'un modelé de détection des objets	35
III.4.1. Créer l'environnement de travail et installer des packages.....	35
III.4.2. Configurer les variables d'environnement PYTHONPATH	39
III.4.3. configuration du modèle et d'entraînement	39
III.4.4. Teste de configuration de TensorFlow	40
III.4.5. Assembler et étiqueter les images	42
III.4.5.1. Assemblément les images	42

III.4.5.2 Étiqueter des images	43
III.4.6. Générer des données d'entraînement	45
III.4.7. Génération et numérotations des objets	46
III.4.8. Création de la carte d'étiquettes et configuration de l'entraînement	47
III.4.8.1. Carte d'étiquettes.....	47
III.4.8.2. Configuration de l'entraînement	48
III.4.9. Organisation de l'entraînement	49
III.4.10. Exporter le graphiques	52
III.4.11. Test et diagnostique de l'état de surface des isolateurs de haute tension par le modèle réalisé.....	53
III.5. Présentation des résultats de l'état de surface des isolateurs	54
III.6. Conclusion	55
 Chapitre IV: La détection d'état de surface d'isolateur par une carte Raspberry Pi	
IV.1. Introduction.....	56
IV.2. Liste des matériel nécessaire	56
IV.3. Modules et toolbox nécessaires.....	59
IV.3.1. Traitement d'image avec open cv	59
IV.3.2. Méthodes de l'objet récoognition	60
V.3.3. Imutls	60
IV.4. Installation de système d'exploitation sur Raspberry pi (Raspbian)	61
IV.5. Exécutez Raspberry Pi et installez des packages	61
IV.6. Entraîner le modèle à reconnaître des objets.....	62
IV.7. Conclusion.....	66
CONCLUSION GENERALE	68
BIBLIOGRAPHIE	69

Introduction générale

INTRODUCTION GENERALE :

Les besoins économiques et le bien-être des populations font partie aujourd'hui des raisons d'un développement croissant des installations de production de transport et de transformation d'énergie électrique [1]. L'approvisionnement de cette énergie électrique provenant des centrales exige l'installation des lignes aériennes et souterraines permettant l'alimentation des consommateurs. Les lignes aériennes qui sont conçues, réalisées, entretenues sont exploitées conformément aux normes, prescription et directives de sécurité, de même qu'aux règles de l'art, sont exposées à diverses contraintes. Parmi celles-ci la pollution des isolateurs [1].

Les isolateurs sont parmi les composants les plus importants des réseaux aériens de transport et de distribution de l'énergie électrique puisqu'ils assurent à la fois le support mécanique des parties portées à la haute tension et l'isolation électrique de ces dernières avec les parties mises à la terre ou énergisées [2]. Premièrement conçus en verre et en céramique, la fabrication des isolateurs a connu, à travers des années, une évolution notable, et ceci dans le but d'accroître leurs performances quelle que soient les conditions météorologiques et environnementales auxquelles ils sont exposés

Le comportement des isolateurs pollués est un sujet d'intérêt croissant dans de nombreux pays. Cet intérêt est lié au désir de choisir au mieux le matériau constituant l'isolateur, étant donné les nombreuses perturbations causées par la pollution [3].

Les contournements électriques des isolateurs entraînent des répercussions graves sur les réseaux aériens de distribution de l'énergie électrique, car ils sont la cause de coupures électriques de durée plus ou moins longues [2]. Depuis une cinquantaine d'années, plusieurs travaux ont été menés pour comprendre les mécanismes conduisant au contournement des isolateurs pollués et se prémunir d'outils permettant la prédiction de ce phénomène [2].

Cette pollution mène à une diminution de la rigidité diélectrique et donc de la résistivité superficielle des matériaux isolants. Par conséquent, des pertes d'énergie, parfois énormes peuvent avoir lieu et conduisent à la mise hors service de la ligne, avec destruction éventuelle de matériels [4]. La tension de contournement d'un isolateur pollué dépend essentiellement de la conductivité des dépôts polluants, c'est à dire du degré de pollution, ainsi que de la répartition de cette couche sur sa surface [5].

Le présent travail exposera, surveillance de l'état de surface des isolateurs de haute tension par l'utilisation d'une technique d'intelligence artificielle

Afin de bien mener notre étude, nous avons subdivisé notre travail en trois chapitres :

Dans le premier chapitre, nous allons parler des différents types des isolateurs des lignes aériennes à la haute tension, ce chapitre présentera aussi définition de phénomène de pollution d'isolateur haute tension et les types de pollution.

Dans le deuxième chapitre nous parlerons des méthodes de diagnostic de l'état des isolateurs et les approches utilisés en détection des objets.

Le troisième chapitre est l'essentielle de ce mémoire, où nous appliquons une technique d'intelligence artificielle pour la détection de l'état de surface d'isolateur de haute tension.

Le quatrième chapitre est la partie amusante de l'étude. Où le modèle de détection d'objets a été créé par Raspberry Pi

Chapitre I :Isolateurs de haute tension et phénomène de pollution

I.1.Introduction:

Dans les lignes de transport d'énergie électrique, l'isolant principal est l'air, cependant, on utilise des isolateurs attachés à des pylônes afin de maintenir les câbles suspendus dans l'air. Dans ce cas, l'isolant joue un rôle essentiellement mécanique. Il n'en demeure pas moins que le matériau utilisé doit posséder d'excellentes propriétés d'isolation électrique. Les matériaux isolants occupent une place importante dans les applications électriques. On les retrouve partout où il y a des conducteurs électriques. Ils doivent être entourés d'isolant [6].

Les lignes aériennes et les postes des réseaux de transport d'énergie électrique sont exposés à diverses contraintes. Parmi celles-ci, la pollution des isolateurs qui constitue l'un des facteurs de première importance dans la qualité et la fiabilité du transport d'énergie. En effet par temps de pluie ou de brouillard, les dépôts polluants se fixant sur les surfaces isolantes réduisent considérablement la résistivité superficielle et le contournement peut alors survenir. L'humidification des couches polluantes facilite en fait, la circulation d'un courant de fuite sur les surfaces isolantes provoquant des échauffements locaux et par la suite l'assèchement de la couche de pollution [7].

Nous présenterons dans ce chapitre une étude sur les isolateurs en précisant leur rôle et leurs différents types, ainsi que la pollution qui représente l'une des contraintes auxquelles sont soumis les isolateurs des réseaux électriques.

I.2.Les isolateurs :

I.2.1. Définition des isolateurs de haute tension :

L'isolateur est un matériau isolant solide qui présente une très grande résistance au passage du courant et dont la conductibilité est pratiquement nulle. Il est utilisé pour isoler les conducteurs ou les pièces sous tension afin d'empêcher les court-circuits, les pertes de courant et les charges d'électrocution.

L'isolateur est constitué principalement d'un matériau isolant solide qui présente une très grande résistance au passage du courant et dont la conductibilité est pratiquement nulle [5]. L'isolateur est utilisé comme son nom l'indique pour l'isolement entre deux corps ou deux pièces sous différentes tensions pour empêcher les courts circuits, les pertes de courant et les charges d'électrocution. L'isolateur est un matériau solide, liquide ou gaz qui a une très grande résistance au passage du courant et dont la conductibilité est pratiquement nulle [8].

Les isolateurs des lignes aériennes ont deux fonctions principales. D'une part, ils permettent d'isoler électriquement les lignes de transport d'énergie électrique aux pylônes qui sont mis à la terre, et d'autre part, ils ont un rôle mécanique qui consiste à soutenir ces mêmes lignes et donc résister aux différentes contraintes mécaniques dû surtout au poids de la ligne, son mouvement en présence de vent... etc. [8].

I.2.2. Fonctionnement et constitution d'un isolateur :

Les isolateurs sont des composants indispensables au transport et à la distribution de l'énergie électrique. Leur fonction est de réaliser une liaison mécanique entre des conducteurs portés à des potentiels différents accrochés aux pylônes des lignes aériennes, ils maintiennent les conducteurs dans la position spécifiée (isolateurs d'alignement et d'ancrage), ils assurent la transition entre l'isolation interne (huile, SF6) et l'isolation externe, ils permettent de raccorder les matériels électriques au réseau (traversées de transformateur, extrémités de Câbles) et ils constituent, également, l'enveloppe de certains appareils (disjoncteurs, parafoudres, réducteurs de mesure).

Les matériaux isolants solides les plus utilisés sont essentiellement le verre trempé et la porcelaine. Mais les résines synthétiques commencent aussi à être utilisées.

Les isolateurs sont conçus et dimensionnés pour résister aux contraintes prévisibles introduites par l'environnement. Du point de vue électrique, l'isolateur est considéré comme étant deux électrodes dont l'intervalle comporte trois zones constituant trois isolants en parallèle ayant des comportements différents, qui sont les suivants :

- ❖ L'intervalle d'air.
- ❖ Le matériau diélectrique.
- ❖ L'interface air-matériau diélectrique [9] (la longueur de l'interface constitue la ligne de fuite ligne le long de laquelle circulerait le courant de fuite).

I.3. Quelques définitions concernant les isolateurs :

I.3.1. Courant de fuite critique :

C'est le courant minimal nécessaire pour provoquer le contournement sous pollution d'un isolateur à tension donnée. Il est indépendant du procédé d'essai ainsi que la forme de l'isolateur

I.3.2. Tension de tenue (Tension d'essai) :

C'est le niveau de tension le plus important, que peut supporter une isolation sans provoquer de décharge disruptive (contournement dans le cas des isolateur) [10]. La tension d'essai ou de tenue est fixée par les règles nationales ou internationales et permet une coordination correcte de l'isolement.

I.3.3. Contournement :

Décharge disruptive le long d'une surface solide dont la trajectoire contourne ce dernier (cas d'isolateur). Le terme contournement est employé pour des décharges disruptives par amorçage dans l'air entourant l'isolateur.

a. Ligne de contournement :

La ligne de contournement (L) est la distance la plus courte dans un gaz entre les électrodes [32].

b. Ligne de fuite :

La ligne de fuite (L_f) est la distance la plus courte le long de la surface d'un isolateur entre les deux électrodes [28].

c. Ligne de perforation :

Distance la plus courte dans la matière isolante d'un isolateur, entre deux parties conductrices.

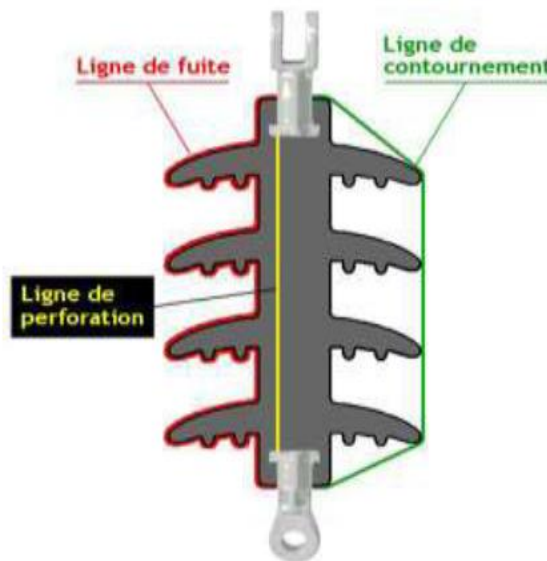


Figure I.1. Les Caractéristiques d'un isolateur.

I.3.4. Contrainte de Contournement :

La contrainte de contournement d'un isolateur est le rapport de tension contournement à la longueur totale de l'isolateur

I.3.5. Tension de Contournement :

La tension de contournement est le niveau de tension le plus bas à partir duquel tous les arcs Oignent les deux électrodes. La tension de contournement dépend [10] :

- ❖ De la résistivité volumique moyenne de la pollution.
- ❖ De la répartition de la couche de pollution.
- ❖ De la longueur de l'isolateur.

I.3.6. Courant de fuite :

C'est un courant de faible amplitude, circulant à travers la couche polluante le long de la surface de l'isolateur, son intensité devient importante lorsqu'on s'approche de la tension de contournement [10]. Il dépend de plusieurs facteurs tels que la nature de la couche polluante et

de la longueur de la ligne de fuite. Dans les conditions de forte humidité, la dissolution des sels contenus dans la couche polluante, provoque la formation d'une couche électrolytique.

I.3.7. Rigidité diélectrique :

C'est la propriété d'un diélectrique de s'opposer à la décharge disruptive. Si on augmente progressivement la tension entre les deux armatures de condensateur jusqu'à une certaine valeur (tension disruptive); il se produit une décharge entre ces deux armatures avec l'apparition d'une étincelle (arc électrique). Numériquement, la rigidité diélectrique est le rapport entre le potentiel appliqué et la distance entre les électrodes au moment où la rupture s'effectue [11].

I.3.8. Résistivité diélectrique :

Sous l'action du champ électrique, toutes les charges se mettent en mouvement. Dans un diélectrique, ce mouvement est un courant électrique, le mouvement des charges libres bien que leur quantité soit très faible constituent un courant de fuite. Ce dernier a deux chemins possibles : il passe à travers le diélectrique lui-même ou bien il s'écoule par la surface du diélectrique. On aura donc un courant de fuite surfacique et un autre volumique. Chacun d'eux développe une résistance surfacique (R_s) et une autre volumique (R_v) [11].

I.3.9. Conductivité électrique :

La conductivité électrique est la capacité d'un matériau à laisser passer et conduire le courant électrique, comme la plupart métaux et les solutions chargées en ions.

I.4.Types d'isolateurs :

Vu les différentes façons d'utilisations et les différentes conditions de fonctionnement des isolateurs, on en distingue des types différents des isolateurs. La différence entre ces types se fait soit par leur fonction (support ou traversée), soit par le matériau isolant utilisé pour la fabrication de ces isolateurs, soit encore par leur profil ou par leur mode de fixation

I.4.1. Isolateurs rigides :

Ce type d'isolateur permet d'obtenir une liaison rigide entre le support et le câble. Il est constitué d'un ou plusieurs isolateurs encastrés et scellés entre eux, ils sont fixés aux pylônes par des ferrures de différentes formes. Ils sont montés sur pylônes soit d'une façon verticale, horizontale ou oblique (inclinée). Mais ce type d'isolateurs est utilisé pour les lignes aériennes qui ne dépassent pas le niveau de tension de 60 KV [12].



Figure I.2. Isolateur rigide.

I.4.2. Isolateurs suspendus ou éléments de chaîne :

Ils sont constitués par le matériau isolant et d'une liaison entre deux isolateurs et à donner une certaine flexibilité à la chaîne d'isolateurs. La pièce métallique qui sert à réaliser la chaîne d'isolateurs est montée sur le pylône en suspension soit verticalement (chaîne d'alignement), soit d'une façon horizontale (chaîne d'ancrage) [13].

Il existe deux types principaux d'éléments de chaîne :

- ❖ L'isolateur capot et tige.
- ❖ L'isolateur long fut.

I.4.3. Isolateur longfut:

Il est constitué d'un bâton cylindrique plein en céramique, en porcelaine ou en matériaux synthétiques muni d'ailettes. A chaque extrémité il y a une pièce métallique pour les liaisons. Cette extrémité métallique peut se présenter sous deux formes distinctes, soit elle enveloppe les extrémités tronconiques sur le cylindre, soit en forme de tige scellée dans une cavité prévue à cet effet [14].



Figure I.3. isolateur long fut.

I.4.4. Isolateur capot et tige :

L'isolateur capot et tige est constitué de trois parties :

- ✓ Bloc isolant ayant à l'intérieur une tige en acier, avec cannelures.
- ✓ Partie supérieure sous forme d'un capot scellé en fonte malléable dans le verre (ou la porcelaine).
- ✓ L'extrémité inférieure de la tige est arrondie formant ainsi la partie inférieure qui a les dimensions adaptées pour pénétrer dans le capot de l'élément suivant, ainsi qu'y rester maintenue par une goupille [17].

Chaque élément est constitué d'un capot, d'une partie isolante en forme de jupe et d'une tige. La coupe d'un tel isolateur est schématisée sur la (Figure I.3) [8]. La forme de la tête est dessinée de sorte que les efforts de traction appliqués à l'isolateur se transforment, autant que possible, en compression des diélectriques sur lesquels apparaissent, toutefois, inévitablement certaines contraintes de cisaillement [15]. Une chaîne d'isolateurs est constituée de plusieurs éléments de type capot et tige ou autres. Ces éléments sont soumis principalement à des efforts de traction. Ils sont généralement utilisés en suspension et forment des chaînes d'isolateurs soit verticales (alignement), soit horizontale (ancrage) [16].

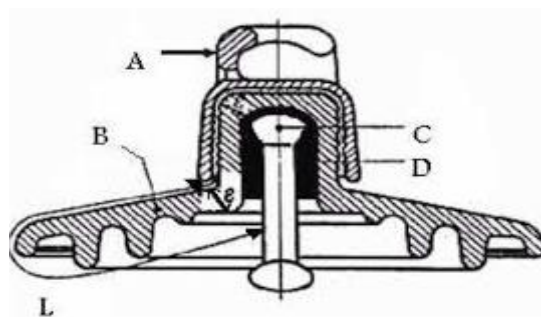


Figure I.4. Isolateurs à capot et tige

A : le capot.

B : le diélectrique (verre trempé ou céramique).

D : le ciment de fixation.

C : la tige.

L : Plus courte distance dans l'air entre le capot et la tige.

ϵ : Longueur du canal de perforation, $\epsilon \ll L/2$.

I.4.4.1. Les différents profils d'Isolateur capot et tige :

Afin d'améliorer le fonctionnement des isolateurs dans les régions polluées, la CEI (comité international d'électrotechnique) a proposé plusieurs profils.

I.4.4.2. Profil standard :

La forme et les dimensions sont en accord avec la normalisation internationale (CEI 305 1978), à cause de leur planéité, les rainures internes bien espacées et la longueur de la ligne de fuite supérieure à la demande standard [18]. Ce type est très utilisé dans les endroits à pollution moyenne



Figure I.5. Isolateur capot et tige avec profil standard

I.4.4.3. Profil anti brouillard (forme A) :

Il est de conception avec un diamètre plus grand que le type de profil standard muni de deux ou trois rainures à grande profondeur. Cette conception permet également un lavage manuel facile si c'est nécessaire et empêche le claquage entre rainures voisines quand il y a une pollution sévère et tout le profil intérieur simplifie la maintenance [19].



Figure I.6. Profil antibrouillard (forme A)

I.4.4.4. Profil antibrouillard (forme B) :

Dans cette conception, l'épaisseur de la rainure extérieure agit comme une barrière contre l'action du brouillard et le dépôt des sels minéraux sur la surface de l'isolateur, empêchant alors la formation d'un électrolyte conducteur sur la surface. Ce type d'isolateur est efficace dans les zones côtières [18].



Figure I.7. Profil antibrouillard (forme B).

I.4.4.5. Profil sphérique :

La forme sphérique permet une longueur de fuite équivalente à celle l'absence de rainures internes permet un lavage manuel facile et efficace [20].



Figure I.8. Profil sphérique

I.4.4.6. Profil plat :

L'élimination complète des rainures internes réduit grandement l'accumulation de pollution sur la surface basse, grâce au courant d'air. Cette conception est particulièrement efficace dans les zones désertiques où l'auto lavage est peu fréquente par pluie [20].



Figure I.9. Profil plat

I.5.Choix des isolateurs :

Les isolateurs entrent pour un pourcentage très modeste de l'ordre de 7%, dans le prix d'une ligne aérienne moyenne tension. Cependant, ils sont un élément essentiel dont dépendent la sécurité d'exploitation, la qualité et la continuité de service.

Les isolateurs les mieux adaptés à un environnement donné sont ceux qui retiennent le taux de dépôts polluants le moins élevé, c'est-à-dire ceux qui possèdent les meilleures propriétés d'auto- nettoyage [16].

Même bien choisie, une isolation n'est jamais à l'abri d'un incident :

- ✓ La sévérité de la pollution d'un site peut changer.
- ✓ L'apparition d'une nouvelle usine à proximité d'un poste.
- ✓ La construction d'un ouvrage routier voisin ou plus simplement, un événement

Météorologique, exceptionnel peuvent augmenter, durablement ou temporairement, la pollution d'un site, alors qu'un poste ou une ligne y sont déjà en exploitation.

Le dimensionnement initialement correct des isolateurs peut alors devenir insuffisant et il faut pouvoir protéger les installations existantes contre les nouvelles sources de pollution éventuelles [17].

I.6. Matériaux isolants utilisés pour la fabrication des isolateurs :

On trouve plusieurs isolants solides qui peuvent être utilisés pour la fabrication des isolateurs de haute tension comme le verre, la céramique et la porcelaine. Mais durant ces dernières années.

La porcelaine est de plus en plus abandonnée à cause de deux inconvénients principaux qui sont : Le poids lourd des isolateurs et la difficulté de détection des amorçages. Ces dernières années on s'intéresse de plus en plus à l'utilisation des isolateurs en matériaux polymères [16].

I.6.1. Céramique :

Le développement et la fabrication des céramiques datent depuis longtemps à cause de leurs performances. Pour les isolateurs qui vont être utilisés dans des lieux où il y a des contraintes mécaniques très importantes, on utilise de préférence des céramiques à grains très fins. Souvent on trouve les céramiques dans les postes : isolateurs supports, couverture isolante des sectionneurs, des disjoncteurs, des transformateurs de potentiel, des bornes de traversées des transformateurs de puissance [16].

I.6.2. Le verre :

Outre son bas prix, le verre présente l'avantage de permettre de déceler les défauts par une simple observation.

On trouve deux types de verre pour la fabrication des isolateurs : le verre trempé et le verre recuit.

a. Le verre trempé :

La résistance à la traction du verre trempé est environ 5 à 6 fois plus grande que le verre recuit et peut supporter des variations brusques de température pouvant atteindre 100° C [16].

b. Le verre recuit :

Le verre recuit est utilisé pour la fabrication d'isolateurs rigides, mais présente cependant des inconvénients. Parmi ces inconvénients, nous citons le fait qu'il ne résiste pas aux variations brusques de température et qu'il ne supporte que des tensions mécaniques relativement faibles et c'est pour cette raison qu'il n'est pas utilisé pour la fabrication des isolateurs de suspension [16].

I.6.3. Matériaux synthétiques :

Les isolateurs en matériaux synthétiques sont composés d'un centre en fibres de verre imprégnées d'une résine et d'un revêtement à ailettes de type élastomère. Leur avantage est qu'ils sont légers et présentent une grande résistance mécanique (Figure I.9).

Ils ont de bonnes propriétés hydrophobes et peuvent être utilisés dans des conditions de pollution très sévères (Figure I.10). L'inconvénient de ces isolateurs est le vieillissement sous l'effet des différentes contraintes auxquelles ils sont soumis (électriques, mécaniques, atmosphériques...) (Figure.I.11) [16].

A : pièce d'accrochage métallique.

B : noyau en fibre de verre-résine.

C : revêtement à ailettes en matériaux synthétiques

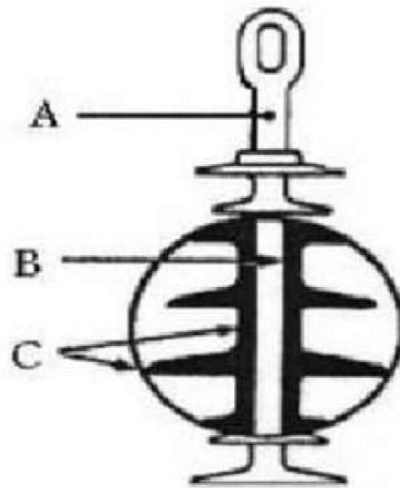


Figure. I.10. Isolateur composite

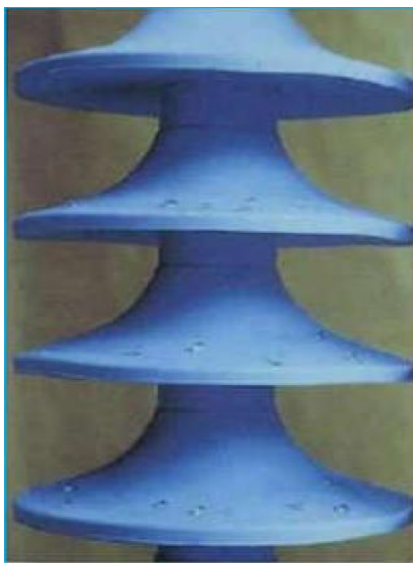


Figure I.11. La prospérité
hydrophobe.



Figure I.12. Vieillissement d'un
isolateur composite

I.7. Défauts des isolateurs :

Les isolateurs composites des générations précédentes ont présenté comme principaux problèmes l'érosion, l'acheminement, la décharge de couronne...etc.

Ces problèmes ont été éliminés par le bon choix des matériaux et le développement de nouvelles techniques de fabrication. L'inconvénient des isolateurs composites est la difficulté d'identifier l'isolateur défectueux de l'isolateur sain, d'où la nécessité de développer une méthode de diagnostic pour différencier entre un isolateur défectueux et un isolateur sain [8].

On peut distinguer deux types de défauts : défauts électriques et défauts mécaniques. Ces deux types de défauts sont mutuellement dépendants.

I.7.1. Défauts électriques :

Le vieillissement des isolateurs est causé par les conditions climatiques auxquelles l'isolateur est exposé ainsi que son profil. Un profil qui permet l'accumulation des agents polluants favorise l'apparition des décharges locales menant à l'érosion et par la suite au cheminement.

L'effet de couronne est considéré comme agent principal de vieillissement des Isolateurs. Cet effet se caractérise par la réduction de l'hydrophobie, la décoloration de la surface de l'isolateur ainsi que l'érosion sous l'effet des gaz de décharge.

Les gouttelettes d'eau sur la surface de l'isolateur affectent la distribution du champ sur cette surface (la permittivité relative de l'eau est élevée ce qui augmente l'intensité du champ aux voisinages des gouttelettes), les gouttelettes se déforment dans la direction du champ.

Un excellent isolateur est caractérisé par son hydrophobicité, cette hydrophobie peut être perdue suite à l'accumulation des agents polluants sur la surface de l'isolant, ou même après avoir subi des chocs électriques (arcs locaux) ; certains isolateurs peuvent réacquérir leurs états hydrophobes après immigration des molécules de faible du centre de l'objet jusqu'à la surface[8].

Dans le cas des isolateurs en verre ou en porcelaine, le mécanisme de décharge est différent de celui des isolateurs en polymère, le mouillage de la pollution accumulée sur la surface des isolateurs en verre ou en porcelaine conduit à l'apparition d'un film d'eau continu, donc l'arc peut se développer directement. Le séchage de la partie restante et l'élongation de l'arc causent le contournement de l'isolateur. Les premières étapes sont similaires à celles des SIR. De plus, la couche de pollution développée est souvent mince, ce qui limite l'intensité du courant de fuite et par la suite, la tension de contournement augmente [21].

I.7.2. Défauts mécaniques :

Les défauts mécaniques d'un isolateur peuvent mener à des conséquences fatales. Dans le cas des isolateurs en SIR, la plupart des défauts sont d'ordre mécanique.

I.7.3. Défauts dus à la manipulation :

Les isolateurs en porcelaine peuvent être endommagés facilement mais le taux de tels endommagements est toujours faible. Les isolateurs composites ne sont pas fragiles mais marquent un taux relativement élevé d'endommagements.

En effet, un nombre d'endommagements plus ou moins important a été constaté dû des manœuvres durant le transport et l'installation. Pour prendre en charge ces problèmes, le groupe CIGRE (Conférence International des Grands Réseaux Electriques) prépara un guide de manipulation qui contient des recommandations relatives aux méthodes de manipulation de ce type d'isolateurs, en commençant de l'endroit de fabrication jusqu'à sa mise sous tension [21].

I.8. Vieillissement des matériaux isolants :

En génie électrique, le vieillissement électrique est défini comme tout changement des propriétés du matériau qui affecte ses fonctions d'isolation. Il peut donc s'agir d'une modification de nature chimique ou physique induisant une modification des propriétés électriques dans un sens défavorable pour la pérennité des fonctions isolantes. Généralement on accepte de façon intuitive, et l'expérience le montre souvent, que la durée de vie d'un matériau isolant dépend de l'intensité de la contrainte électrique subie par celui-ci mais également de la durée d'application de cette contrainte [4].

I.8.1. Farinage :

Apparition à la surface du matériau synthétique des particules de charge formant une surface rugueuse ou poudreuse.

I.8.2. Faïençage :

Micro fractures superficielles peu profondes de (0,01 à 0,1 mm).

I.8.3. Craquelures :

Fractures superficielles de profondeur supérieure à 0,1 mm.

I.8.4. Erosion :

Dégradation irréversible et non conductrice, qui peut se produire, à la surface de l'isolateur, par perte de matière. Elle peut être uniforme, localisée, ou arborescente. Après des amorçages partiels, il peut se produire sur les isolateurs composites des traces superficielles, en générale arborescentes, comme sur les pièces en porcelaine, tant qu'elles ne sont pas conductrices, ces traces ne sont pas préjudiciables. Quand elles sont conductrices, elles se classent dans le phénomène d'acheminement.

I.8.5. Acheminement :

Dégradation irréversible, par formation de chemins qui apparaissent et se développent à la surface d'un matériau isolant. Les chemins sont conducteurs même à sec. L'acheminement peut se produire sur des surfaces au contact de l'air et aussi aux interfaces entre matériaux isolants différents .il est en général matérialisé par l'apparition d'une trace de matériau carbonisé.

I.9. Pollution des isolateurs :

Par définition, le mot « pollution » désigne la dégradation d'un milieu qui se produit par l'introduction, directe ou indirecte, de substances nocives pour l'environnement ou par la modification de ses caractéristiques biologiques, chimiques ou physiques. Cela implique que, lorsqu'on parle des isolateurs, le phénomène de pollution est défini comme étant la dégradation de la surface isolante par des électrolytes solubles ou des particules inertes non solubles, qui sont des substances ayant la capacité d'altérer la performance électrique des isolateurs [22].

Le terme « pollué », souvent associé avec le mot « sale » au sens usuel, prend ainsi une autre signification lorsqu'il est utilisé pour décrire l'état d'un isolateur. Autrement dit, un isolateur pollué n'a pas nécessairement l'apparence d'être sale. En effet, un isolateur qui est fortement pollué par un embrun salin peut paraître propre alors que sa performance électrique subit une forte dégradation. Par contre, un autre isolateur recouvert de suie, qui paraît sale, peut avoir une performance électrique remarquable. D'où, un isolateur est considéré comme d'augmenter la conductivité surfacique de cet isolateur. Le mot « contamination » est alors plus approprié dans ce cas pour décrire le phénomène de pollution des isolateurs et il se réfère plutôt à la conductivité électrique de la surface des isolateurs qu'à leur apparence extérieure [22].

Même bien choisie, une isolation n'est jamais à l'abri d'un incident. La sévérité de la pollution d'un site peut changer. L'apparition d'une nouvelle usine à proximité d'un poste, la construction d'un ouvrage routier voisin ou plus simplement, un événement météorologique exceptionnel peuvent augmenter, durablement ou temporairement, la pollution d'un site, alors qu'un poste ou une ligne y sont déjà en exploitation. Le dimensionnement initialement correct des isolateurs peut alors devenir insuffisant et il faut pouvoir protéger les installations existantes contre les nouvelles sources de pollution [16].

I.9.1. Sources de pollution :

Les substances polluantes qui contaminent les isolateurs peuvent provenir de différentes sources dépendamment du type d'environnement dans lequel se trouve le site. À titre d'exemples, dans un environnement de type marin, les électrolytes solubles proviennent des embruns salins de la mer qui sont présents dans l'atmosphère et constituent ce qu'on appelle la pollution marine. Dans un environnement type industriel, les isolateurs sont exposés à la pollution industrielle dont les substances solubles proviennent de la fumée produite par des centrales électriques thermiques (à charbon ou au mazout), des aciéries ou des raffineries alors que les substances inertes non solubles proviennent de la poussière générée par des cimenteries ou des mines de ciment. Et particulièrement en hiver, la plupart des isolateurs qui sont installés à proximité des autoroutes sont sujets à la pollution hivernale dont les sources d'électrolytes solubles sont principalement le salage des routes.

Il arrive parfois que la fumée provenant du chauffage domestique dans un milieu urbain à forte densité de population constitue aussi une source de pollution en hiver. Certes, il existe d'autres types d'environnement qui sont également susceptibles de produire les substances polluantes selon la norme IEC 60815 (2008) [22].

I.9.1.1 Types de pollution :**I.9.1.1.1 Pollution naturelle :**

La pollution naturelle provient [23] :

- ✓ Des sels marins dans les régions côtières.
- ✓ Des poussières du sol (notamment lors de chantiers importants).
- ✓ Des sables véhiculés par le vent en régions désertiques.

I.9.1.1.1.1. Pollution marine :

Dans les installations en bordure de mer, les embruns pores par le vent déposent progressivement sur les isolateurs une couche de sel qui, à plus ou moins longue échéance, recouvre toute la surface de l'isolateur y compris les parties les mieux protégées. Cette couche de sel, humidifiée par les embruns eux même, ou par un brouillard ou simplement par condensation, devient conductrice. Un courant de fuite s'établit alors à travers la couche superficielle et des arcs électriques peuvent prendre naissance dans certaines conditions, et ce d'envelopper jusqu'à provoquer le contournement total de l'isolateur [24].

I.9.1.1.1.2. Pollution désertique :

Dans les régions désertiques, les fréquentes tempêtes de sables déposent progressivement sur les surfaces des isolateurs une couche de pollution contenant des sels. Une fois humidifiée, cette couche devient beaucoup plus conductrice [16]. Et engendrent la circulation d'un courant de fuite qui apparaît brusquement et qui est suivi par l'apparition d'arcs partiels qui peuvent conduire au contournement total de l'isolateur [23].

I.9.1.1.2. Pollution industrielle :

Ce type de pollution est fréquent dans les zones industrielles surtout à proximité des usines, entreprises de production éjectant des fumées (raffinerie, cimenterie,). Par ailleurs les gaz d'échappements (centrale thermiques) et les engrais utilisés en agriculture contribuent aussi aux dépôts observés à la surface des isolateurs [25] [26].

En présence d'une humidité intense, le sel contenu dans ces polluants abaisse considérablement la résistivité superficielle des isolateurs et le contournement peut parfois avoir lieu [27].

Les usines ne sont pas les seules responsables de ce genre de pollution ; les gaz d'échappement des véhicules et les engrais utilisés en agriculture contribuent aussi aux dépôts observés à la surface des isolateurs [23].

I.9.1.1.3. Pollution mixte :

Ce type de pollution est en fait le plus fréquent et le plus sévère pour l'exploitation des ouvrages électriques. La pollution mixte résulte de la combinaison entre les différents types de pollution, comme par exemple les pollutions marine et industrielle lorsque des installations industrielles situées au bord de mer [23].

I.9.1.2. Classifications de types de pollution :

La composition de cette pollution est variable suivant les sources de contamination et les conditions auxquels sont soumis les isolateurs. La norme CES 815 donne le classement de la pollution suivant son origine (Tableau I-1). Essentiellement, les types de pollution qui existent sont : la pollution naturelle, la pollution industrielle et la pollution mixte [28].

Tableau I.1. Classification de la pollution selon le type d'environnement

Niveaux de pollution	Exemples d'environnements typiques
I. Faible	- Zones sans industries et avec faible densité d'habitations équipées d'installations de chauffage. - Zones avec faible densité d'industries ou d'habitations mais soumises fréquemment aux vents et/ou pluies. - Régions agricoles. - Régions- montagneuses.
II. Moyen	- Zones avec industries ne produisant pas de fumées particulièrement polluantes et/ou avec une densité moyenne d'habitations équipées d'installations de chauffage. - Zones à forte densité d'habitations et/ou d'industries mais soumises fréquemment aux vents et/ou à des chutes de pluies. - Zones exposées au vent de mer, mais pas trop proches de la côte.
III. Fort	- Zones avec forte densité d'industries et banlieues de grandes villes avec forte densité d'installation de chauffage polluant. - Zones situées près de la mer, ou en tout cas exposées à des vents relativement forts venant de la mer.
IV. Très fort	- Zones généralement peu étendues, soumises à des poussières conductrices et à des fumées industrielles- produisant des dépôts-conducteurs particulièrement épais. - Zones généralement peu étendues, très proches de la côte et exposées aux embruns et aux vents très forts et polluants venant de la mer. - Zones désertiques caractérisées par de longues périodes sans pluie, exposées aux vents forts transportant du sable et du sel et soumise à une condensation régulière.

I.9.2. Formation et répartition des couches polluantes :

Suite à l'écoulement d'un air transportant des poussières diverses, une couche de pollution se forme à la surface des isolateurs. La répartition non-uniforme et non-homogène de cette couche dépend du profil de l'isolateur, de la position de la chaîne par rapport au sol (verticale, horizontale, inclinée), du niveau de tension ainsi que du degré de pollution du site où est situé l'isolateur [8].

La non-uniformité de la pollution peut être classée en trois catégories :

a) *Non-uniformité longitudinale par groupe :*

Elle est caractérisée par un ensemble de groupes d'isolateurs, soumis à différentes conductivités de la couche de pollution, dont la valeur est constante dans chaque groupe.

Nous retrouvons ce type de pollution dans les cas suivants :

- temporairement pendant le lavage sous tension.
- dans les chaînes d'isolateurs en forme de « T ».
- par effet du champ électrique où la concentration de la pollution est accentuée sur les isolateurs les plus proches de la borne haute tension [29].

b) Non-uniformité transversale :

Ce type est caractérisé par des secteurs ou bandes de différentes conductivités superficielles de la couche polluante. Ces bandes sont réparties transversalement autour de la surface de chaque isolateur de la chaîne. La conductivité dans chaque secteur est la même le long de la ligne de fuite.

Ce type de pollution est dû principalement à l'existence d'une direction privilégiée des vents et des pluies [29].

c) Non-uniformité longitudinale périodique :

Ce type est le plus fréquent. Il est caractérisé par une variation périodique de la conductivité de la couche de pollution le long de la ligne de fuite, mais elle garde une symétrie circulaire. Ces principales spécifications sont :

- la face inférieure de l'isolateur présente une conductivité plus grande que la face supérieure.
- la concentration de la pollution augmente de la zone périphérique vers la zone centrale.
- la pollution est plus accentuée entre les nervures [29].

I.9.3. Conséquences de la pollution :

Les couches polluantes qui s'accumulent à la surface des isolateurs provoquent une conductivité électrique superficielle [30]. Celle-ci modifie la répartition du potentiel le long de la ligne de fuite. Suivant les conditions atmosphériques (pluie fine, brouillard, ...), la tension de rupture diélectrique de l'air peut être atteinte entre deux points de la surface isolante entraînant l'amorçage d'un arc électrique qui court-circuite une partie de la ligne de fuite [16].

Trois cas peuvent se présenter selon les contraintes auxquelles est soumis l'isolateur :

a) Arc non localisé :

On appelle arc non fixe (non localisé), lorsqu' il s'amorce et s'éteint rapidement pour se Réamorcer dans un autre endroit sur la surface d'un isolateur [6]. Il y a apparition de courant de fuite entraînant une petite perte d'énergie, généralement supportable par l'installation [31].

b) Arcs fixe :

Contrairement au précédent, il se fixe sur la surface de l'isolateur, soit en s'y maintenant (pour un courant continu), soit en s'y réamorçant au même endroit (courant alternatif). Ce type d'arcs peut provoquer une dégradation de l'isolation par son effet thermique [6].

c) Contournement des isolateurs :

Le contournement d'une surface polluée et humidifiée, dans ce cas, l'arc électrique atteint l'électrode opposée [6]. Ce qui entraîne l'apparition d'un courant de fuite accompagné de la formation d'une bande sèche et d'arcs partiels, et la propagation de l'arc qui peut couvrir tout l'isolateur [25]. Ce phénomène conduit à une rupture de l'isolation.

I.10. Conclusion :

A travers ce chapitre on conclut que pour remplir leur rôle, les matériaux isolants et les isolateurs, doivent vérifier des critères mécaniques, électriques et environnementaux. Chaque type d'isolateurs est destiné pour une isolation précise. Ainsi le choix de l'isolateur, ou le dimensionnement même de celui-ci, se fait en tenant compte des critères (contraintes) déjà mentionnés pour assurer une qualité et continuité de service, aussi réaliser un couple qualité prix optimal [6].

La pollution des isolateurs est un facteur essentiel dont il faut tenir compte dans la conception des lignes électriques de haute tension. Pour mieux dimensionner l'isolateur, il est indispensable de connaître la sévérité de la pollution des sites concernés. La connaissance de cette sévérité consiste à étudier les différents paramètres qui définissent l'état de dégradation de l'isolation [16].

La tension de contournement d'un isolateur pollué dépend essentiellement de la conductivité du dépôt polluant (degré de pollution) ainsi que de la répartition de la couche de pollution sur la surface isolante. La connaissance de ce degré de pollution est par conséquent, une condition préalable et indispensable pour apprécier le niveau de l'isolement des ouvrages installés sur site, en vue de dimensionner convenablement l'isolation [33].

Chapitre II :

Méthodes de diagnostic et de détection des objets

II.1. Introduction :

Les isolateurs de type extérieurs sont soumis à diverses contraintes d'exploitations et d'environnement. La contamination de la surface des isolateurs constitue l'un des facteurs majeurs qui augmente la probabilité du contournement. Au cours des deux dernières décennies, une variété de techniques se basant sur le traitement du signal du courant de fuite a été proposée afin de suivre le comportement de l'isolateur pollué en temps réel. Parmi ces techniques, nous citons la transformée de Fourier, la transformée en ondelette, le technique récurrent plot et d'autres. Des méthodes d'apprentissage supervisé ont été également exploitées par plusieurs chercheurs dans le but de prédire le degré de sévérité de la pollution déposée sur la surface des isolateurs [34].

Détection un objet dans une image est une opération qui consiste à distinguer la région objet de celle du fond suivant un critère défini. Suivre un objet dans une séquence d'images est une opération qui consiste à localiser la région objet. Plusieurs techniques peuvent être utilisées afin d'assurer ces opérations [35].

Dans ce chapitre, nous présentons une synthèse des plus récents travaux portant sur la pollution ainsi que le diagnostic des isolateurs pollués. Aussi, nous présentons les techniques les plus importantes utilisées pour détection d'un objet.

La détection d'objets et de formes se fait classiquement par la mise en correspondance de points d'intérêt. Cependant, depuis 2012 d'autres méthodes plus directes mais difficilement interprétables (apprentissage profond), ont révolutionné le domaine en présentant des résultats bien supérieurs. Cependant, nous avons pu montrer [36,37] que les points d'intérêt d'une image peuvent se calculer en appliquant des filtres orientés à chaque pixel de l'image et que ces filtres peuvent fournir aussi des descripteurs. Les mises en correspondance obtenues surpassent très nettement les méthodes classiques. Il serait alors possible d'appliquer ces nouveaux filtres en évitant un apprentissage profond.

II.2. Les méthodes de diagnostic :

II.2.1. Définition :

An sens, strict, ces méthodes supposent qu'aucun modèle n'est disponible pour décrire les relations de cause à effets. La seule connaissance repose sur l'expertise humaine confortée par un solide retour d'expérience. Dans cette catégorie, on retrouve toutes les méthodes basées sur l'intelligence artificielle, la reconnaissance des formes, les systèmes experts et les réseaux de neurones artificiels [33].

II.2.2. Prédiction du degré de sévérité de pollution :

Dans le but de déterminer l'influence de la sévérité de la couche de pollution sur le Comportement diélectrique des chaines d'isolatrices hautes tensions, nous avons utilisé une base de données récupérée au niveau du laboratoire de haute tension [34].

Ces essais ont été effectués sur une chaine d'isolateur réel U120B.

Dans ce chapitre on décrit en premier lieu les techniques expérimentales utilisées pour récupérer les signaux du courant de fuite. Pour cela plusieurs essais préliminaires ont été réalisés dans le but de s'assurer le bon fonctionnement du matériel en tenant compte de toutes les contraintes qu'elles soient physiques ou techniques [34].

En deuxième lieu, nous présentons et discutons les résultats concernant la prédiction du degré de sévérité de pollution déposée sur la surface isolante de la chaîne en se basant sur l'analyse du signal du courant de fuite correspondant à chaque cas de pollution [34].

II.2.3. Enregistrement du signal du courant de fuite :

Nous montrons dans cette partie l'équipement et la chaîne d'isolateur utilisée ainsi que les différentes techniques expérimentales permettant l'enregistrement du signal du courant de fuite.

Le laboratoire de haute tension utilisé est muni d'une station d'essai à fréquence industrielle (50 Hz) de marque HEAFELY composée par les équipements suivants :

- ✓ Un transformateur d'essai : 500V/300kV, 50kVA, 50Hz.
- ✓ Un transformateur de réglage : 0/220V à 500V, 50kVA, 50Hz.
- ✓ Un diviseur de tension capacitif qui est relié au secondaire du transformateur d'essai pour réduire la tension aux bornes du voltmètre, situé près du pupitre de commande. Il est composé d'une capacité $C1=400\ \mu\text{F}$ et d'une capacité variable qui permet de régler la tension de sortie.
- ✓ Des appareils de mesures et de protection.
- ✓ Un oscilloscope numérique « TEKTRONIX » de bande passante 500 MHz.
- ✓ Une caméra numérique SONY DCR-SR45 pour suivre l'évolution des décharges électriques en fonction de la tension appliquée et de la distribution de la pollution sur l'isolateur étudié.
- ✓ Un ordinateur pour l'acquisition des données des signaux du courant de fuite équipé du programme labVIEW [34].

II.2.4. Origines de LabVIEW :

Il a été créé en 1986, initialement pour Apple Macintosh, qui était à l'époque l'un des seuls ordinateurs proposant une interface graphique native. L'histoire de LabVIEW explique un vocabulaire spécifique, et explique encore certaines actions.

À l'origine, LabVIEW s'exécute sur des écrans noir et blancs, puis sur des écrans 16 couleurs, 256, etc. LabVIEW est un environnement de développement propriétaire (par opposition à un environnement ouvert, permettant à plusieurs personnes de développer des compilateurs compatibles, comme pour C/C++, Ada, Java, Fortran, etc.) développé et vendu par la société National Instruments (NI) [34].

Le premier métier de NI est de fabriquer du matériel d'acquisition (notamment sur le protocole GPIB au début des années 80) rapidement destiné au marché des microordinateurs (IBM PC, Apple Macintosh). Ainsi, la première version de LabVIEW s'attache à offrir un

environnement de développement dont le rôle est de permettre simplement à l'utilisateur de créer des instruments virtuels (Virtual instrument, ou vi) utilisant le matériel d'acquisition NI pour reproduire sur un micro-ordinateur le comportement d'un instrument personnalisé et personnalisable à volonté [34].

II.2.5. Utilité de LabVIEW :

Le principal intérêt de LabVIEW est de permettre facilement de développer ces interfaces à l'aide ensemble de codes utilisant un langage de programmation graphique qui permet de contrôler les objets de la face avant. Dans un certain sens, le diagramme se trouvant sur la face arrière ressemble à un flow charte [34].

II.2.6. Initiation aux instruments virtuels de LabVIEW :

Les programmes LabVIEW sont appelés instruments virtuels, ou Vis, car leur apparence et leur fonctionnement s'apparentent aux instruments réels, tels que les oscilloscopes et les multimètres. LabVIEW contient une grande gamme d'outils pour l'acquisition, l'analyse, l'affichage et l'enregistrement des données, ainsi que des outils pour vous aider à mettre au point votre programme [34].

Dans LabVIEW, on construit une interface utilisateur, ou face-avant, avec des commandes et des indicateurs. Les commandes sont des boutons rotatifs, des boutons-poussoirs, des cadrans et autres mécanismes d'entrée. Les indicateurs sont des graphes, des LED et d'autres afficheurs de sortie. Après avoir construit l'interface utilisateur, vous ajoutez du code en utilisant des Vis et des structures pour contrôler les objets de la face-avant. Le diagramme contient ce code[34]

On peut utiliser LabVIEW pour communiquer avec le matériel comme, par exemple, d'acquisition de données ou d'images, les périphériques de commande d'axes et les instruments GPIB, PXI, VXI, RS232 et RS485. et pour obtenir et visualiser le signal du courant de fuite sur l'isolateur on a dû établir un montage virtuel de la figure II.1 en utilisant les instrument de ce programme [34].

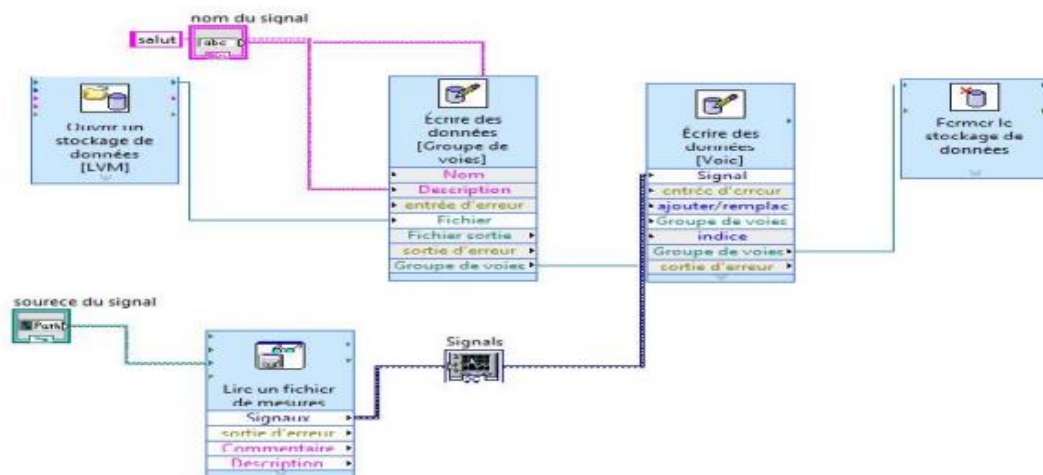


Figure II.1 : le schéma de montage virtuel pour visualiser un signal sur LabVIEW

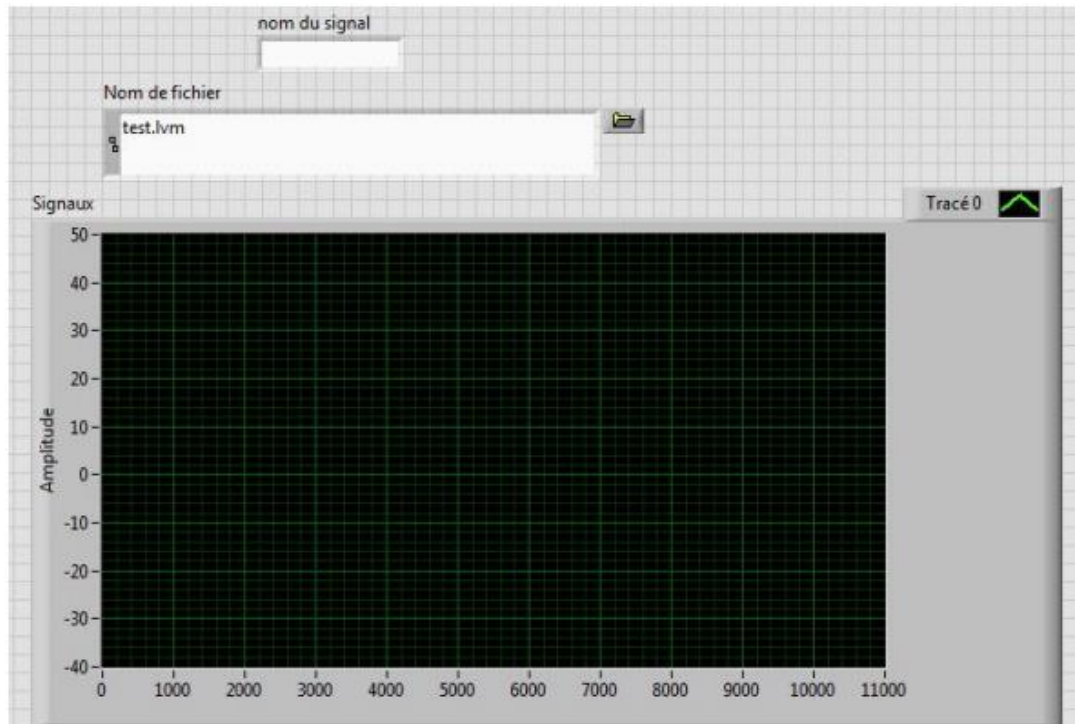


Figure II.2. Afficheur de l'oscilloscope virtuel sur Labview

Le circuit d'essai ainsi que le dispositif expérimental sont donnés dans la figure suivante.

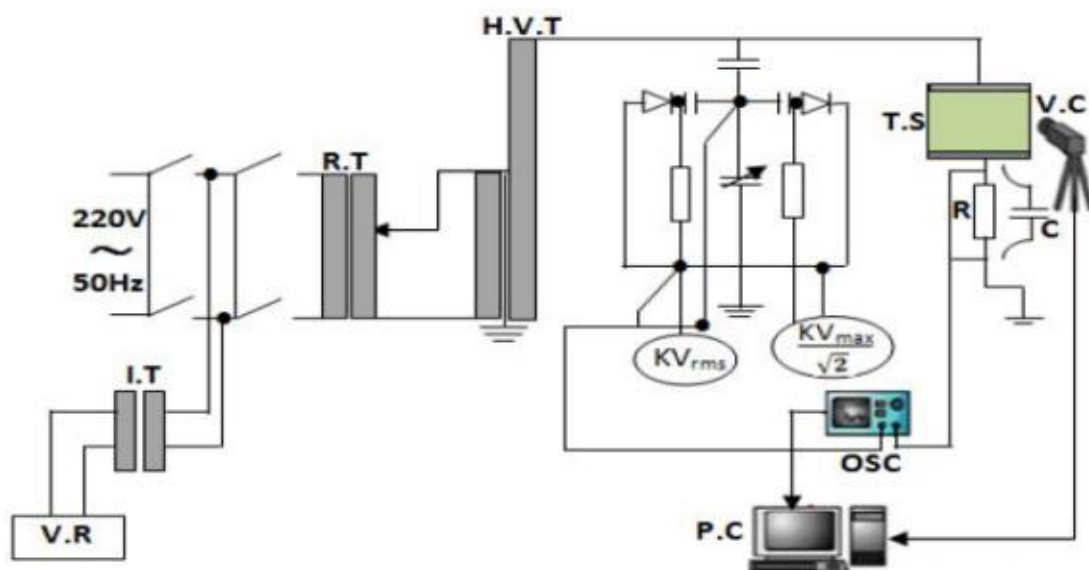


Figure II.3 : Dispositif d'essais et de mesure

II.3. Isolateurs testés :

L'U 120 B caractérisé par une résistance mécanique de 120 kN, un diamètre de 25.5 cm et une ligne de fuite de 30 cm par élément. Les tensions de tenue à fréquence industrielle à sec et sous pluie sont respectivement 60 et 45 kV. Le choix de cet isolateur est justifié par la forme qu'il présente. Cette dernière nous permet aisément d'appliquer la pollution d'une façon

uniforme ou bien non-uniforme. La chaîne d'isolateurs réels U120B sur laquelle les expériences ont été effectuées est montrée sur la figure (II.4). [34]



Figure II.4 : Chaîne d'isolateurs U120B étudiée

II.4 Procédures de test :

II.4.1. Préparation des chaînes d'isolateurs :

Le nettoyage des chaînes d'isolateurs a été effectué après chaque essai. En premier lieu, l'eau de robinet est utilisée pour éliminer le maximum d'impuretés, puis l'isolateur est séché avec des serviettes en papier. Un deuxième nettoyage est effectué à l'aide d'un bout de coton imbibé d'alcool chirurgical à 70° afin d'assurer une propreté quasi-totale de l'isolateur. Ce dernier est laissé sécher naturellement dans le laboratoire [34].

II.4.2. Préparation des solutions polluantes :

Les solutions polluantes ont été préparées au laboratoire du Génie de l'Environnement. Celles-ci sont composées de différentes quantités de kiselghur et NaCl diluées dans un litre d'eau distillée. La préparation des solutions polluantes ainsi que le choix de leurs conductivités volumiques sont faites en se basant sur les recommandations de la CEI 60060-1.

Selon le profil de l'isolateur étudié, nous choisissons les solutions polluantes ainsi que leurs conductivités volumiques [34].

Les solutions polluantes sont composées de 100 g de kiselghur dilué dans un litre d'eau distillée pour lesquelles on rajoute des quantités pesées de NaCl. Les quantités de NaCl rajoutées sont 5, 10, 20, 30 et 40 g. Celles-ci ont été pesées à l'aide d'une balance dont la précision est de 10^{-4} g [34].

II.4.3. Méthode d'application de la pollution :

La couche de pollution est déposée sur la surface de l'isolateur réel par le procédé de pollution liquide pré-déposée. Pour cela, un pinceau de largeur égale à 8 cm est utilisé afin que la couche de pollution soit déposée d'une façon cohérente possible sur la surface de l'isolateur [34].

La figure suivante montre les deux surfaces interne et externe d'un élément de la chaîne d'isolateur [34].



Figure II.5 : Surfaces interne et externe d'un élément de la chaîne d'isolateur

II.5 Circuits de mesure :

II.5.1. Mesure de la tension de contournement :

La mesure de la tension de contournement a été effectuée pour chaque configuration de pollution à l'aide du voltmètre électrostatique, donnant la valeur de la tension en kilovolts efficace. Cette mesure permet non seulement de déterminer l'impact de la configuration de la pollution sur la tension de contournement, mais aussi de déterminer les paliers des tensions à appliquer, pour l'enregistrement du courant de fuite [34].

II.5.2. Mesure du courant de fuite :

Pour toutes les configurations de pollution étudiées, les mesures du courant de fuite ont été réalisées via un oscilloscope numérique. Pour évaluer le courant de fuite, on récupère le signal de la tension aux bornes d'une résistance insérée en série entre l'objet d'essai et le retour de la terre. Le signal ainsi récupéré représente l'image du courant de fuite. Tous les signaux ont été enregistrés avec une fréquence d'échantillonnage de 100 kHz et les données sont stockées dans un ordinateur pour une exploitation ultérieure. Chaque test est répété entre six et dix fois pour vérifier la reproductibilité des conditions initiales [34].

II.6 Exploitation du signal du courant de fuite pour la prédiction de la sévérité de Pollution :

Dans ce chapitre, nous nous intéressons à l'étude des harmoniques du signal du courant de fuite. Pour cela, nous calculons l'énergie des harmoniques les plus significatifs (1^{er} , $3^{ème}$, $5^{ème}$, $7^{ème}$ et $9^{ème}$) à l'aide de l'analyse spectrale [34].

Le calcul de l'énergie des harmoniques du courant de fuite est effectué pour cinq conductivités différentes à savoir 1.2 - 3.5 - 6 - 10 et 20 mS/cm. Nous rappelons également que la pollution est déposée d'une manière non-uniforme sur l'isolateur [34].

L'objectif de cette étude est de comprendre d'une part, la cause des distorsions observées sur les signaux du courant de fuite et d'autre part, d'évaluer la quantité d'énergie du courant de fuite qui est responsable de l'évaporation de l'eau en présence des décharges électriques à la surface de l'isolateur. Notre but est de diagnostiquer la sévérité de pollution en fonction des harmoniques du courant de fuite ayant une énergie significative [34].

II.6.1 L'analyse spectrale :

Pour analyser l'onde du courant de fuite, notre choix s'est porté sur le spectre d'amplitude qui est calculé à partir de la FFT du signal du courant de fuite, ce choix offre une bonne résolution fréquentielle donc une distinction facile des différents harmoniques. La figure ci-dessous illustre la décomposition d'un signal en spectre d'amplitudes [34].

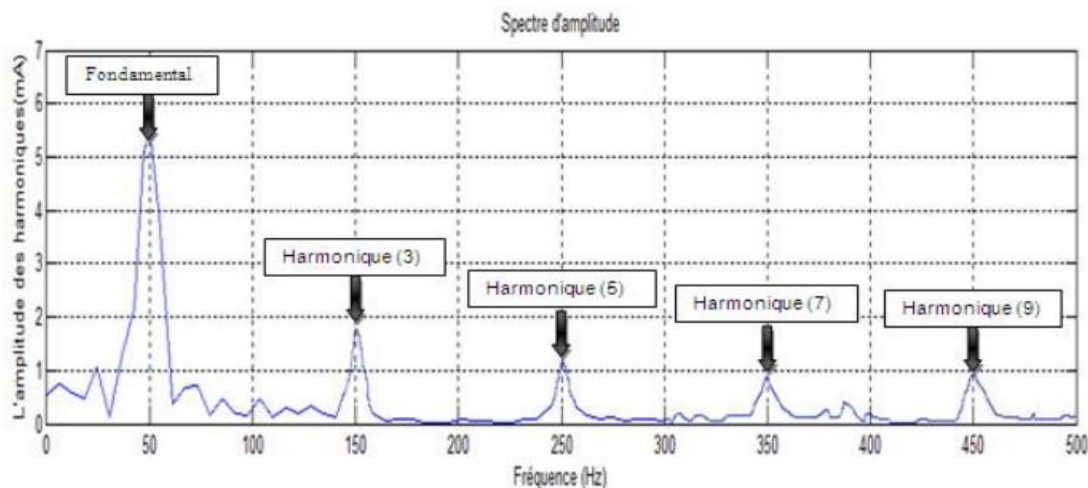


Figure II.6 : Représentation du spectre d'amplitude.

II.6.2 Application de l'analyse spectrale aux signaux du courant de fuite :

Les figures suivantes représentent les signaux du courant de fuite ainsi que les spectres d'amplitudes correspondant dans l'évolution du processus de contournement d'un isolateur haute tension sous pollution non-uniforme pour les différentes conductivités étudiées et pour trois degrés de pollution distincts [34].

La figure II.7 montre le signal du courant de fuite pour une faible pollution, nous remarquons que le signal est pratiquement sinusoïdal, ne comportant aucune distorsion indiquant l'activité des décharges électriques sur la surface de l'isolateur [34].

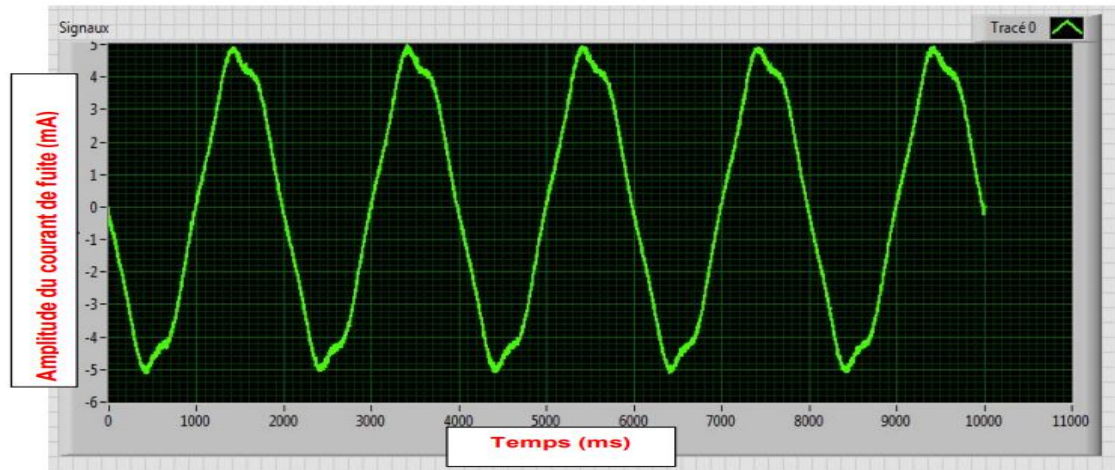


Figure II.7. Représentation du signal du courant de fuite correspondant à une faible pollution

Pour le signal représentant la moyenne pollution (figure II.8) nous notons l'occurrence des distorsions sur l'onde du courant de fuite. Ces distorsions sont beaucoup plus localisées sur les sommets des deux alternances. Ces distorsions sont dues probablement à l'activité moderne des décharges partielles sur la surface du modèle plan [34].

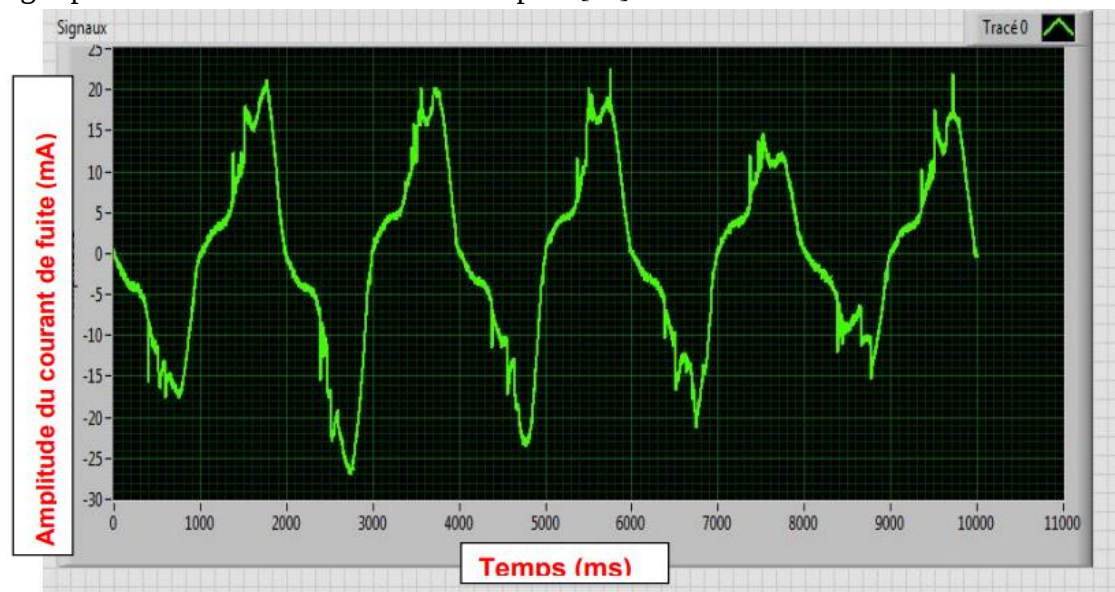


Figure II.8. Représentation du signal du courant de fuite correspondant à une moyenne pollution

Le signal enregistré pour la plus forte pollution (figure II.9) comporte pratiquement des distorsions sur la totalité de l'onde du courant de fuite. Dans ce cas, caractérisé par une activité intense des décharges électriques (intensification et propagation), le signal du courant de fuite ne possède plus sa forme sinusoïdale. Nous observons aussi des pics sur ce signal indiquant le moment de l'occurrence des décharges électriques. Une analyse attentive de ces signaux montre que la valeur du courant de fuite prend une valeur constante durant un moment [34].

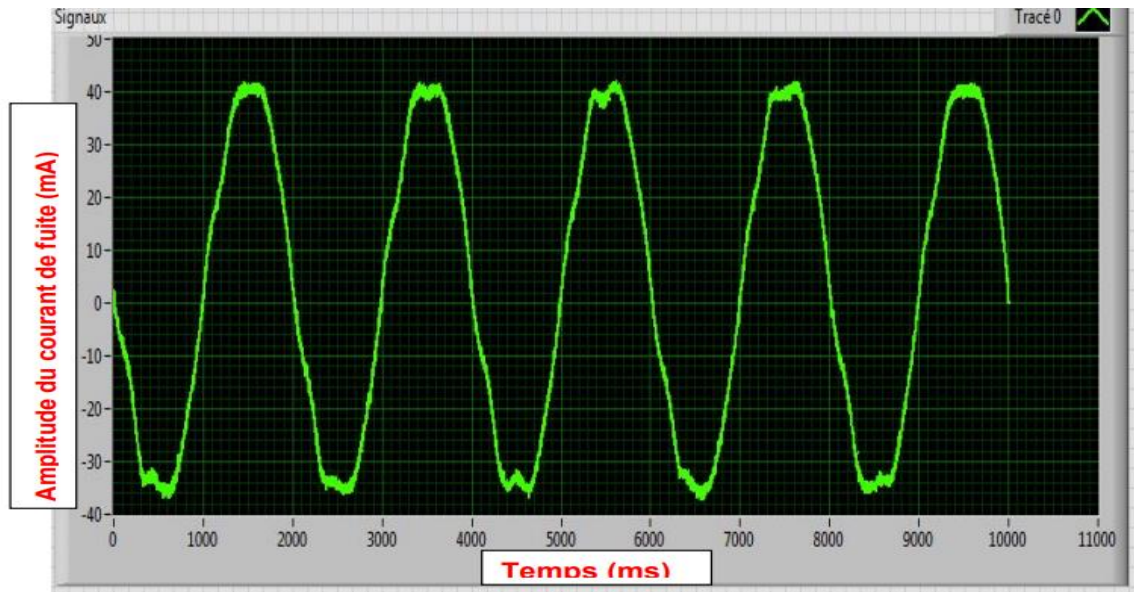


Figure II.9. Représentation du signal du courant de fuite correspondant à une forte pollution

II.7. Méthodes de détection des objets :

II.7.1. Définition :

La détection d'objets est un domaine très actif de la recherche qui cherche à classer et localiser des régions/zones d'une image ou d'un flux vidéo. Ce domaine est à la croisée de deux autres : la classification d'image et la localisation d'objets. En effet, le principe de la détection d'objets est le suivant : pour une image donnée, on recherche les régions de celle-ci qui pourraient contenir un objet puis pour chacune de ces régions découvertes, on l'extrait et on la classe à l'aide d'un modèle de classification d'image -par exemple-. Les régions de l'image d'origine ayant de bons résultats de classification sont conservés et les autres jetés. Ainsi, pour avoir une bonne méthode de détection d'objets, il est nécessaire d'avoir un algorithme solide de détection de régions et un bon algorithme de classification d'images. [35] Localiser un objet dans une image est complexe et mesurer la performance de la localisation requiert une métrique adaptée. De plus, au contraire de la classification, il peut y avoir plusieurs objets dans la même image et le détecteur doit pouvoir le préciser (Figure II.11) [36].



Figure II.10. Exemple de détection d'objets

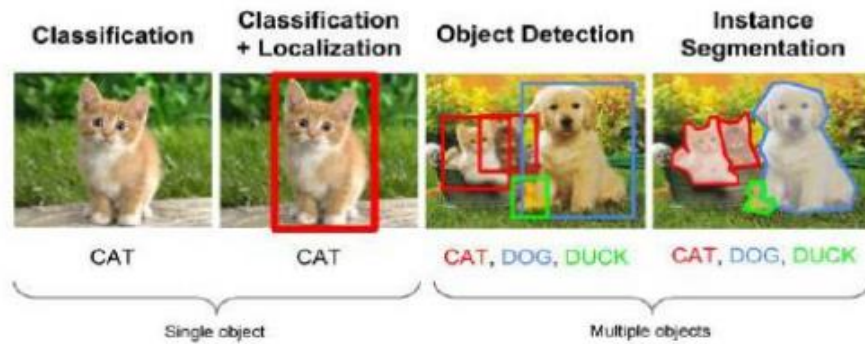


Figure II.11. Illustration de la différence entre la classification et la détection d'objets

II.7.2. OpenCV

II.7.2.1. Définition OpenCV :

OpenCV (Open Source Computer Vision) est une bibliothèque libre de vision par ordinateur. Cette bibliothèque est écrite en C et C++ et peut être utilisée sous Linux, Windows et Mac OS X. Des interfaces ont été développées pour Python, Ruby, Matlab et autre langage. Open CV est orienté vers des applications en temps réel. Un des buts d'OpenCV est d'aider les gens à construire rapidement des applications sophistiquées de vision à l'aide d'infrastructure simple de vision par ordinateur. La bibliothèque d'OpenCV contient plus de 500 fonctions [36].

II.7.2.2. Fonctionnalités :

La bibliothèque OpenCV met à disposition de nombreuses fonctionnalités très diversifiées permettant de créer des programmes partant des données brutes pour aller jusqu'à la création d'interfaces graphiques basiques. Elle propose la plupart des opérations classiques en traitement bas niveau des images et des vidéos [36].

II.7.2.2.1. Traitement d'images :

Elle propose la plupart des opérations classiques en traitement bas niveau des images :

- ✓ Lecture, écriture et affichage d'une image.
- ✓ Calcul de l'histogramme des niveaux de gris ou d'histogrammes couleurs.
- ✓ Lissage, filtrage.
- ✓ Seuillage d'image (méthode d'Otsu, seuillage adaptatif).
- ✓ Segmentation (composantes connexes, GrabCut).
- ✓ Morphologie mathématique [36].

II.7.2.2.2. Traitement vidéo :

Cette bibliothèque s'est imposée comme un standard dans le domaine de la recherche parce qu'elle propose un nombre important d'outils issus de l'état de l'art en vision des ordinateurs tels que :

- ✓ Lecture, écriture et affichage d'une vidéo (depuis un fichier ou une caméra).

- ✓ Détection de droites, de segment et de cercles par Transformée de Hough.
- ✓ Détection de visages par la méthode de Viola et Jones.
- ✓ Cascade de classifieurs boostés.
- ✓ Détection de mouvement, historique du mouvement.
- ✓ Poursuite d'objets par mean-shift ou Camshift.
- ✓ Détection de points d'intérêts.
- ✓ Estimation de flux optique (Méthode de Lucas–Kanade).
- ✓ Triangulation de Delaunay.
- ✓ Diagramme de Voronoi.
- ✓ Enveloppe convexe.
- ✓ Ajustement d'une ellipse à un ensemble de points par la méthode des moindres carrés [36].

II.7.2.3. Modules OpenCV :

OpenCV a une structure modulaire. Les principaux modules d'OpenCV sont énumérés ci-dessous :

a. *CxCore (Cœur d'OpenCV) :*

Contient les structures de donnée et les opérations mathématiques de base. Elle permet de réaliser les fonctions d'algèbre matriciel, la transformation de données, la gestion de mémoire, la manipulation d'erreurs, le chargement dynamique de code et permet aussi la réalisation des graphiques [36].

b. *HighGui (bibliothèque d'interface graphique):*

OpenCV intègre sa propre bibliothèque haut-niveau pour ouvrir, enregistrer et afficher des images et des flux vidéo. Celle-ci contient aussi un certain nombre de fonctions permettant de réaliser des interfaces graphiques très simples, mais largement suffisantes pour tester nos programmes [36].

c. *ImgProc (traitement d'image) :*

Ce module comprend des algorithmes de traitement d'image de base, y compris le filtrage d'image, transformations d'image, les conversions d'espace couleur et etc... [36].

d. *Vidéo (traitement de flux vidéo):*

Ceci est une vidéo module d'analyse qui inclut des algorithmes de suivi (tracking) d'objets, des algorithmes de soustraction de fond et etc... [36]

e. *ObjDetect (détection d'objets)*

Cela comprend l'objet de détection et de reconnaissance des algorithmes pour les objets standards [36].

f. *ML (bibliothèque d'apprentissage automatique)*

Comporte les fonctions de classification, d'analyse de donnée et des outils de clustering [36].

g. Calib3D

Calibrage caméra et reconstruction 3D [37].

II.7.3. Tensorflow :

II.7.3.1. Définition :

TensorFlow est un outil qui permet d'implémenter et de tester rapidement des algorithmes d'apprentissage artificiel profonds. Une de ses caractéristiques est d'être adapté à des architectures matérielles dédiées au calcul (GPU) et de permettre une distribution de ce calcul sur des grappes de machines très simplement. Il permet d'utiliser de nombreux algorithmes prêts à l'usage [38]. Elle a été utilisée pour la recherche et le déploiement de l'apprentissage automatique dans une grande variété de domaines, contenant la reconnaissance vocale, la vision par ordinateur, la robotique, la récupération d'informations, le traitement du langage naturel, etc... [39].

II.7.3.2. Fonctionnement :

Le fonctionnement interne de TensorFlow est la clé de son succès. Un calcul TensorFlow est décrit par un graphe, qui se compose d'un ensemble de nœuds. Le graphe représente un flux de données et ses transformations, avec des extensions permettant à certains types de nœuds de maintenir et de mettre à jour des états persistants. Les différents clients (Python, C++, autres) permettent de générer ce type de graphe de calcul. Un exemple de code construisant un graphe de calcul puis l'exécutant est visible dans la Figure II.12. [39].

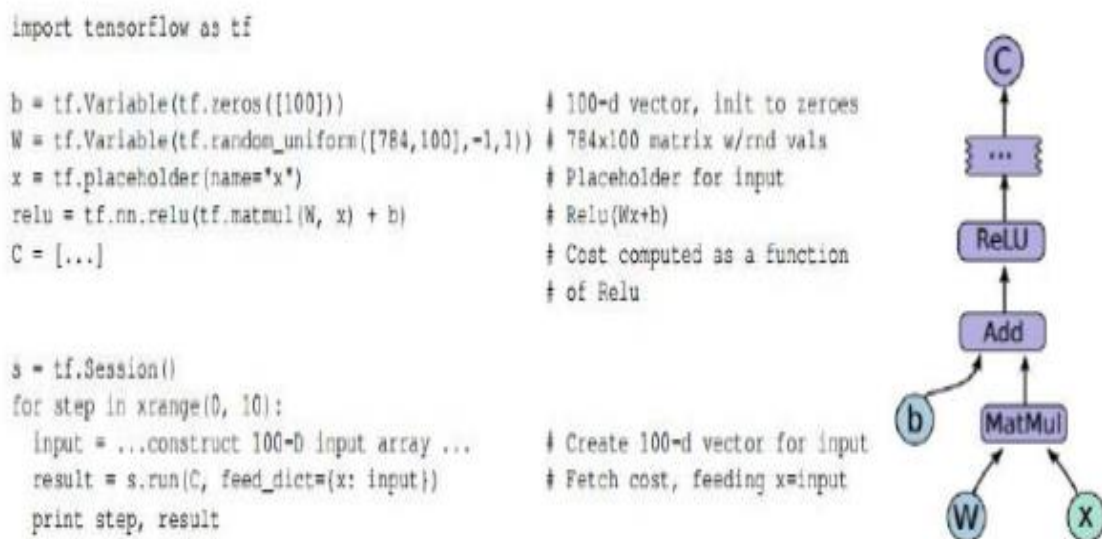


Figure II.12. Exemple de génération de graphe de calcul à partir de code (Tensorflow)

II.7.3.3. Les composants :

Les composants de TensorFlow sont :

- ✓ Le graphe de calcul, qui contient l'information de toutes les opérations qu'il est possible d'effectuer
- ✓ Chaque nœud correspond à une opération possédant zéro ou plusieurs entrées et de même pour la sortie.
- ✓ Une arête définit le sens suivant lequel circulent les tenseurs dans le graphe
- ✓ Un tenseur est un tableau de dimensionnalité non définie mais de type connu ou inféré depuis le graphe.
- ✓ Une opération représente un calcul abstrait (par exemple, multiplication matricielle, ou addition). Elle possède un nom et des attributs éventuellement pour son exécution (exemple type des tenseurs)
- ✓ Un noyau est une implémentation particulière d'une opération exécutée sur un type particulier de périphérique (par exemple CPU ou GPU).
- ✓ Un binaire TensorFlow définit les opérations du graphe et leurs noyaux associés et disponibles [36].

Les programmes peuvent interagir avec le système TensorFlow en créant une Session durant laquelle un est appelé définissant les noms de variables à calculer ainsi qu'un ensemble de tenseurs optionnels pour alimenter les entrées éventuelles du graphe. En utilisant ces arguments, l'implémentation de TensorFlow peut calculer tous les nœuds qui doivent être exécutés afin de calculer les sorties demandées, et peut alors organiser leur exécution. [36]

La connaissance exacte et complète du graphe de calcul a de multiples impacts. Le premier est la facilitation de la parallélisations ou distribution de toutes ces opérations sur plusieurs supports du type CPU ou GPU ou entre machines. En effet, en déterminant la répartition entité/opération, il est facile de préparer les communications entre machines et les noyaux à utiliser suivant les supports. La deuxième conséquence est la possibilité de faire du calcul symbolique et donc de faire de la différenciation automatique. Si le gradient de chaque opération élémentaire est défini, il est possible de remonter automatiquement au gradient de chaque variable du graphe grâce à la règle de la chaîne. Et comme évoqué précédemment, cette différenciation automatique est un facteur primordial pour faire du l'apprentissage en profondeur et créer de nouvelles structures [36].

II.8. Conclusion :

Cette étude nous a permis de conclure que l'énergie des harmoniques du courant de fuite est corrélée avec la sévérité de la pollution. Pour la pollution non-uniforme l'occurrence et la propagation des décharges électriques sur la surface de l'isolateur est détectée par l'augmentation des énergies des harmoniques en fonction de leur rang. La distorsion du signal du courant est détectée en grande partie par l'évolution des différents harmoniques. Nous avons aussi pu constater que l'occurrence des décharges électriques accentue la distorsion du signal du courant, à cause des chutes de tension observées au niveau de la crête.

Les techniques actuelles les plus performantes pour évaluer le degré de sévérité de la pollution des isolateurs sont fondées pratiquement sur l'application des méthodes de traitement de signal sur l'onde du courant de fuite ou les images ou le champ. La synthèse bibliographique des différents travaux antérieurs effectuée dans le présent chapitre nous permet de nous inspirer sur l'application de ces méthodes dans le domaine de la haute tension et spécialement dans celui de la pollution des isolateurs.

L'application de ces méthodes qui constituent un outil très efficace pour la prédiction du comportement de l'isolateur en service sous les contraintes de pollution. Rappelons aussi que ces méthodes peuvent être combinées avec des méthodes d'apprentissage supervisé afin de prédire également le degré de sévérité de pollution.

Peuvent être utilisés les méthodes des détections des objets pour diagnostiquer l'état de surface d'isolateurs. Où il est un outil très efficace pour la prédiction de l'état de surface d'isolateur.

Chapitre III : Détection de l'état des surface d'isolateur par les techniques de TensorFlow

III.1. Introduction :

La détection d'objets est un terme général pour les technologies de vision par ordinateur pour localiser et identifier des objets. Les techniques de détection d'objets peuvent être appliquées à la fois aux images fixes ou aux images en mouvement. Les technologies de vision par ordinateur sont déjà largement aujourd'hui utilisées dans le monde, car elles peuvent fournir des informations précises sur l'état des isolateurs individuels ou de la chaîne entière et aider les employés pour suivre l'état de l'isolateur et visualiser leur surface pendant la mise en service. L'utilisations des méthodes de détections d'objets dans le domaine de haute tension ont pour but de déterminer l'influence de la sévérité de la couche de pollution sur le comportement diélectrique des chaines des isolateurs de hautes tensions.

Parmi le processus des développements technologiques, l'introduction des méthodes méta-heuristiques est devenue une nécessité dans tous les domaines de la vie et fait partie de la transmission de l'énergie électrique.

Dans ce chapitre, notre objectif est d'élaborer un modèle qui capable a détecté l'état de surface des isolateurs de haute tension par les techniques d'intelligence artificielle. Dans cette étude, nous utilisons la méthode TensorFlow dans l'interface de programmation python.

III.2. Abstraite :

La découverte d'objets est largement utilisée dans le monde et ses utilisateurs, y compris le personnel enseignant, les industriels et les passionnés de robot, utilisent des classificateurs basés sur les réseaux de neurone en conjonction avec d'autres techniques de détection d'objets. Le but de cette étude est d'explorer les solutions modernes basées sur l'open source pour la détection d'objets dans le réseau électrique, en l'occurrence pour la détection d'isolateurs de haute tension. L'API *TensorFlow Object Detection*, est une Framework avec une source ouverte pour les tâches liées à la détection d'objets, elle a été utilisée pour entraîner et tester un SSD (Single-shot Multiple Box Detector) à l'aide d'un modèle *Mobilenet*. Le modèle a été testé comme pré-entraîné et affinage d'un ensemble de données constitué d'images extraites depuis des séquences vidéo des isolateurs de haute tension.

Les hypothèses examinées avant l'utilisation de la méthode de détection d'objet :

- 1)Le modèle pré-entraîné ne fonctionnera pas pour les données non traitées.
- 2)Un formulaire précis fonctionnera raisonnablement bien sur les données fournies.
- 3)L'utilisation des données d'entraînement plus variables améliorera les résultats sur les nouvelles images.

III.3. Modules et toolbox nécessaires :

Les bibliothèques open source telles que la bibliothèque DNN d'OpenCV et l'API TensorFlow Object Détection fournissent des domaines de travail open source faciles à utiliser où les modèles de détection d'objets pré-entraînés (tels que ceux qui peuvent être téléchargés à

partir du modèle zoo TensorFlow) atteignent une grande précision dans la détection de divers objets provenant des isolateurs de haute tension dans le réseau électrique [2]. Cependant, les séquences vidéo utilisées dans cette étude ont été tournées avec des caméras, de sorte que la vue était sous un angle élevé, et les isolateurs dans les images sont souvent très petits ou partiellement cachés par d'autres éléments. Il était donc important de former les modèles à travailler avec des données spécifiques. Dans cette étude, l'API de détection d'objets TensorFlow a été choisie en raison de sa popularité pour la détection d'objets et de sa nature facile à aborder.

Pour cette raison, nous utilisons un ordinateur puissant et plus rapide pour peuvent exploiter les logiciels, modules et toolbox suivantes.

Tableau III.1. Modules et toolbox nécessaires

Nombre	1	2	3	4	5	6	7
Les exigences	Windows10	Anaconda3	Python 3.6	TensorFlow v1.12	CUDA Toolkit v8.0	CuDNN v6.0	LabelImg
Symbole							

Nous avons aussi besoin d'installer plusieurs packages dans l'environnement anconda3.

III.4. Processus de création d'un modelé de détection des objets :

III.4.1. Création de l'environnement de travail et installation des packages :

Afin de créer un environnement qui inclut le fonctionnement de ce modèle, nous ouvrons la fenêtre d'invitation Anaconda puis nous créons le dossier de travail par la commande suivant : `conda create -n projet pip python=3.6`

```
(base) C:\Users\user>conda create -n projet pip python=3.6
WARNING: A conda environment already exists at 'C:\Users\User\anaconda3\envs\projet'
Remove existing environment (y/[n])? y
Collecting package metadata (current_repodata.json): done
Solving environment: done

--> WARNING: A newer version of conda exists. <--
  current version: 4.9.2
  latest version: 4.10.0
Please update conda by running
  $ conda update -n base -c defaults conda

## Package Plan ##
  environment location: C:\Users\user\anaconda3\envs\projet
  added / updated specs:
    - pip
    - python=3.6

The following NEW packages will be INSTALLED:

certifi                pkgs/main/win-64::certifi-2020.12.5-py36haa95532_0
pip                    pkgs/main/win-64::pip-21.0.1-py36haa95532_0
python                 pkgs/main/win-64::python-3.6.13-h2758d61_0
setuptools              pkgs/main/win-64::setuptools-52.0.0-py36haa95532_0
sqlite                  pkgs/main/win-64::sqlite-3.35.4-h2bbff1b_0
vc                      pkgs/main/win-64::vc-14.2-h21ff451_1
vs2015_runtime          pkgs/main/win-64::vs2015_runtime-14.27.29016-h5e58377_2
wheel                   pkgs/main/noarch::wheel-0.36.2-pyhd3eb1b0_0
wincertstore            pkgs/main/win-64::wincertstore-0.2-py36h7fe50ca_0

Proceed ([y]/n)? y
```

Figure III.1. Création de l'environnement de travail.

Après nous confirmons la création en cliquant sur le bouton y, l'environnement a été activé en premier, nous activons l'environnement avec le code suivant [activate projet](#), puis nous installons les packages bureautiques nécessaires selon l'ordre du tableau suivant :

Tableau III.2. Code d'installation des packages

Nombre	Package	Code d'installation
1	Tensorflow==1.12	C:\> Conda install tensorflow=1.12
2	Anaconda protobuf	C:\> conda install -c anaconda protobuf
3	Pillow	C:\> pip install pillow
4	Lxml	C:\> pip install lxml
5	Cython	C:\> pip install Cython
6	contextlib2	C:\> pip install contextlib2
7	Jupyter	C:\> pip install jupyter
8	Matplotlib	C:\> pip install matplotlib
9	Pandas	C:\> pip install pandas
10	OpenCV-python	C:\> pip install opencv-python

Après avoir terminé l'installation des Packages, nous confirmons leurs installations conforme par le code suivant :

[Conda list](#)

```

Select Anaconda Prompt (anaconda3)
# packages in environment at C:\Users\user\anaconda3\envs\projet:
#
# Name                        Version      Build      Channel
tensorflow-select            2.2.0        eigen
absl-py                      0.9.0        py36_0
argon2-cffi                  20.1.0       pypi_0     pypi
astor                        0.8.0        py36_0
attrs                        19.3.0       pypi_0     pypi
backcall                     0.2.0        pypi_0     pypi
blas                          1.0          mkl
bleach                       3.1.5        pypi_0     pypi
ca-certificates              2021.4.13    haa95532_1
certifi                      2020.12.5    py36haa95532_0
cffi                         1.14.1       pypi_0     pypi
colorama                     0.4.3        pypi_0     pypi
contextlib2                  0.6.0.post1  pypi_0     pypi
cyclor                       0.10.0       pypi_0     pypi
cython                       0.29.21      pypi_0     pypi
decorator                    4.4.2        pypi_0     pypi
defusedxml                   0.6.0        pypi_0     pypi
entrypoints                  0.3          pypi_0     pypi
gast                         0.3.3        py_0
grpcio                       1.27.2       py36h351948d_0
h5py                         2.10.0       py36h5e291fa_0
hdf5                         1.10.4       h7ebc959_0
icc_rt                       2019.0.0     h0cc432a_1
importlib-metadata           1.7.0        pypi_0     pypi
intel-openmp                 2020.1       216
ipykernel                    5.3.4        pypi_0     pypi
ipython                      7.16.1       pypi_0     pypi
ipython-genutils             0.2.0        pypi_0     pypi
ipywidgets                   7.5.1        pypi_0     pypi
jedi                         0.17.2       pypi_0     pypi
jinja2                       2.11.2       pypi_0     pypi
jsonschema                   3.2.0        pypi_0     pypi
jupyter                      1.0.0        pypi_0     pypi
jupyter-client               6.1.6        pypi_0     pypi
jupyter-console              6.1.0        pypi_0     pypi
jupyter-core                 4.6.3        pypi_0     pypi
keras-applications           1.0.8        py_1
keras-preprocessing          1.1.0        py_1
kiwisolver                   1.2.0        pypi_0     pypi
libprotobuf                  3.12.3       h7bd577a_0
lxml                          4.5.2        pypi_0     pypi

```

(A)

markdown	3.1.1	py36_0	
markupsafe	1.1.1	pypi_0	pypi
matplotlib	3.3.4	pypi_0	pypi
mistune	0.8.4	pypi_0	pypi
mkl	2020.1	216	
mkl-service	2.3.0	py36hb782905_0	
mkl_fft	1.1.0	py36h45dec08_0	
mkl_random	1.1.1	py36h47e9c7a_0	
nbconvert	5.6.1	pypi_0	pypi
nbformat	5.0.7	pypi_0	pypi
notebook	6.1.3	pypi_0	pypi
numpy	1.15.4	pypi_0	pypi
numpy-base	1.15.4	py36hc3f5095_0	
object-detection	0.1	pypi_0	pypi
opencv-python	4.4.0.40	pypi_0	pypi
openssl	1.1.1k	h2bbff1b_0	
packaging	20.4	pypi_0	pypi
pandas	1.1.0	pypi_0	pypi
pandocfilters	1.4.2	pypi_0	pypi
parso	0.7.1	pypi_0	pypi
pickleshare	0.7.5	pypi_0	pypi
pillow	7.2.0	pypi_0	pypi
pip	20.2.2	py36_0	
prometheus-client	0.8.0	pypi_0	pypi
prompt-toolkit	3.0.6	pypi_0	pypi
protobuf	3.12.3	py36h33f27b4_0	anaconda
pycparser	2.20	pypi_0	pypi
pygments	2.6.1	pypi_0	pypi
yparsing	2.4.7	pypi_0	pypi
pyreadline	2.1	py36_1	
pyrsistent	0.16.0	pypi_0	pypi
python	3.6.10	h9f7ef89_2	
python-dateutil	2.8.1	pypi_0	pypi
pytz	2020.1	pypi_0	pypi
pywin32	228	pypi_0	pypi
pywinpty	0.5.7	pypi_0	pypi
pyzmq	19.0.2	pypi_0	pypi
qtconsole	4.7.5	pypi_0	pypi
qtpy	1.9.0	pypi_0	pypi
scipy	1.5.0	py36h9439919_0	
send2trash	1.5.0	pypi_0	pypi
setuptools	49.2.0	py36_0	
six	1.15.0	py_0	

(B)

slim	0.1	pypi_0	pypi
sqlite	3.32.3	h2a8f88b_0	
tensorboard	1.12.2	py36h33f27b4_0	
tensorflow	1.12.0	eigen_py36h67ac661_0	
tensorflow-base	1.12.0	eigen_py36h45df0d8_0	
termcolor	1.1.0	py36_1	
terminado	0.8.3	pypi_0	pypi
testpath	0.4.4	pypi_0	pypi
tf-slim	1.1.0	pypi_0	pypi
tornado	6.0.4	pypi_0	pypi
traitlets	4.3.3	pypi_0	pypi
vc	14.1	h0510ff6_4	
vs2015_runtime	14.16.27012	hf0eaf9b_3	
wcwidth	0.2.5	pypi_0	pypi
webencodings	0.5.1	pypi_0	pypi
werkzeug	1.0.1	py_0	
wheel	0.34.2	py36_0	
widgetsnextension	3.5.1	pypi_0	pypi
wincertstore	0.2	py36h7fe50ca_0	
zipp	3.1.0	pypi_0	pypi
zlib	1.2.11	h62dcd97_4	

(C)

Figure III.2. (A) (B) (C) List de Package .

III.4.2. Configurer les variables d'environnement PYTHONPATH :

les variables de PYTHONPATH doivent être créées dans les répertoires \models, \models \research et \models \research \slim. Pour ce faire, nous exécutons les commandes suivantes dans l'interface Anaconda :

```
(projet)C:\>setPYTHONPATH=C:\ projet \models;C:\ projet \models\research;C: \ projet  
\models\research\slim
```

III.4.3. configuration du modèle et d'entraînement :

Ensuite, il faut compiler les fichiers Protobuf, qui sont utilisés par TensorFlow pour configurer notre modèle et les paramètres d'entraînement. Chaque fichier racine de TensorFlow d'extension proto dans le répertoire \object_detection\ protos doit être appelé par les commandes individuellement.

Dans l'invite de commande Anaconda, nous exécutons les commandes nécessaires pour ouvrir les répertoires qui contiennent les packages de TensorFlow et ensuite nous créons les fichiers proto dans les répertoires correspond; ce travail se fait par les commandes suivantes :

```
C:\> cd
```

```
C:\tensorflow1\models\research
```

```
protoc --python_out=. .\object_detection\protos\anchor_generator.proto  
.\object_detection\protos\argmax_matcher.proto  
.\object_detection\protos\bipartite_matcher.proto .\object_detection\protos\box_coder.proto  
.\object_detection\protos\box_predictor.proto .\object_detection\protos\eval.proto  
.\object_detection\protos\faster_rcnn.proto  
.\object_detection\protos\faster_rcnn_box_coder.proto  
.\object_detection\protos\grid_anchor_generator.proto  
.\object_detection\protos\hyperparams.proto .\object_detection\protos\image_resizer.proto  
.\object_detection\protos\input_reader.proto .\object_detection\protos\losses.proto  
.\object_detection\protos\matcher.proto .\object_detection\protos\mean_stddev_box_coder.proto  
.\object_detection\protos\model.proto .\object_detection\protos\optimizer.proto  
.\object_detection\protos\pipeline.proto .\object_detection\protos\post_processing.proto  
.\object_detection\protos\preprocessor.proto  
.\object_detection\protos\region_similarity_calculator.proto  
.\object_detection\protos\square_box_coder.proto .\object_detection\protos\ssd.proto  
.\object_detection\protos\ssd_anchor_generator.proto  
.\object_detection\protos\string_int_label_map.proto .\object_detection\protos\train.proto  
.\object_detection\protos\keypoint_box_coder.proto  
.\object_detection\protos\multiscale_anchor_generator.proto
```

[.\object_detection\protos\graph_rewriter.proto](#) [.\object_detection\protos\calibration.proto](#)
[.\object_detection\protos\flexible_grid_anchor_generator.proto](#)

Après la succession des commandes précédentes, la création du fichier "name_pb2.py" à partir de chaque fichier "name.proto" dans le dossier "\ object_detection \ protos" est terminer.

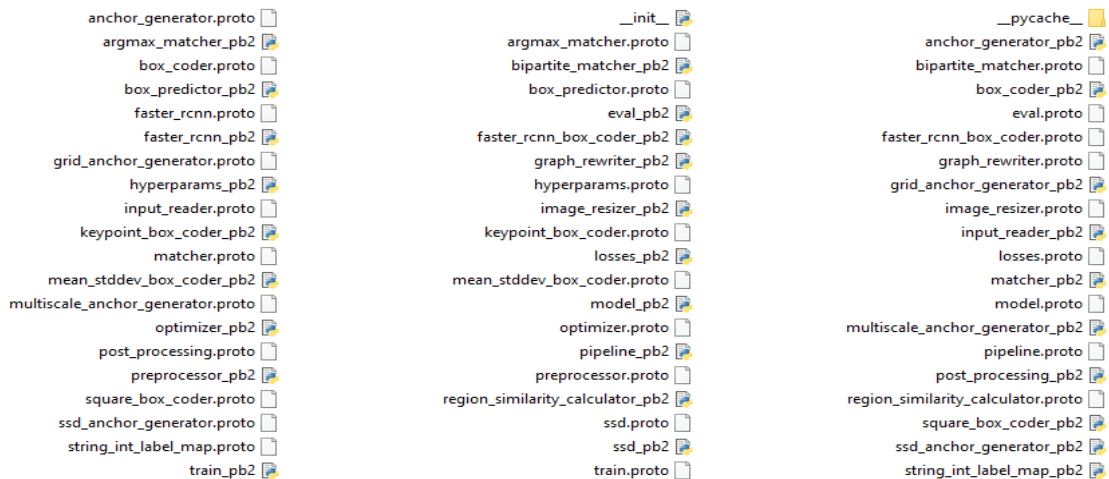


Figure III.3. Création des fichier name_pb2

Enfin, nous pouvons exécuter les commandes suivantes à partir du répertoire C:\tensorflow1\models\research:

(projet) C:\tensorflow2\models\research> python setup.py build

Cette commande crée un dossier build basé sur les ordres de setup. Puis leur installation se fait par la deuxième commande :

(projet) C:\tensorflow2\models\research> python setup.py install

III.4.4. Teste de configuration de TensorFlow :

L'API de détection d'objets TensorFlow est configurée pour utiliser des modèles pré-entraînés pour la détection d'objets ou pour le développement d'un nouveau modèle. Le teste et la vérification de la fonctionnement d'installation ce fait par l'exécution d'un script du package Jupyter. Cette vérification assurée par la commande suivante:

(projet) C:\tensorflow2\models\research\object_detection> jupyter notebook
object_detection_tutorial.ipynb

Cela ouvre le script par défaut dans le navigateur Web et permet d'exécuté les codes à installés. nous pouvons installés chaque section en cliquant sur le bouton «Exécuter» dans la barre d'outils (figure III.3).

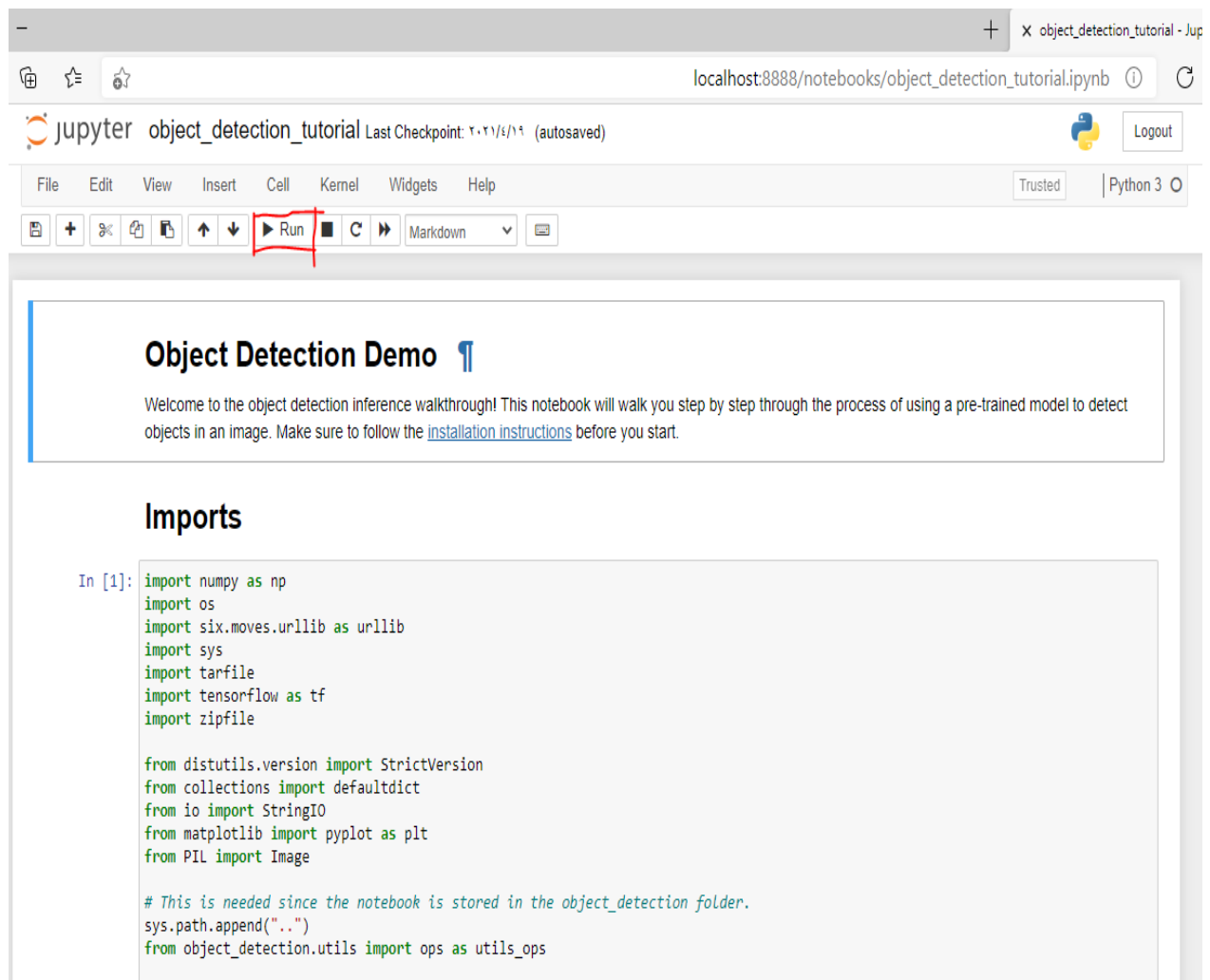


Figure III.4. Installation des packages Jupiter notebook pour la détection des objets

Une fois que nous parcourons le script, nous devons voir les deux images légendées dans la section inférieure de la page. Si nous obtenons cela, alors tout fonctionne correctement ! Sinon, la section inférieure signalera toutes les erreurs rencontrées.

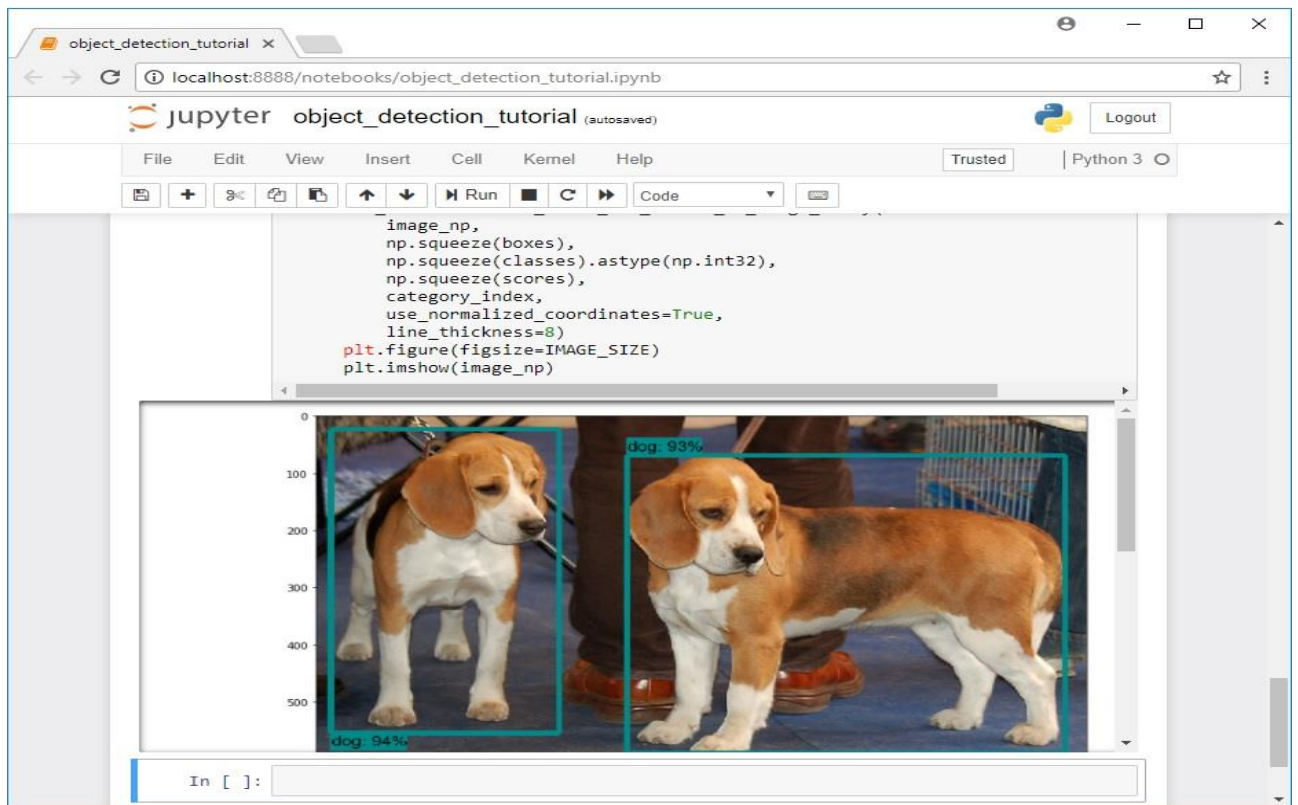


Figure III.5. Teste de l'installation des packages Jupyter notebook.

III.4.5. Assembler et étiqueter les images:

Cette fois, l'API de détection d'objets TensorFlow est configurée et prête à l'emploi, nous devons fournir les images qu'elle utilisera pour former un nouveau classificateur de détection.

III.4.5.1. Assemblage des images :

TensorFlow a besoin des centaines d'images de l'objet qui doit être détecté pour former un bon classificateur de détection. Pour former un classificateur robuste, les images de formation doivent avoir des objets aléatoires dans l'image avec les objets souhaités, et doivent avoir une variété d'arrière-plans et de conditions d'éclairage. Il devrait y avoir des images où l'objet souhaité est partiellement obscurci, chevauché avec autre chose ou seulement à mi-chemin de l'image

Pour notre classificateur de détection d'isolateur, nous avons 2 objets différents à détecter

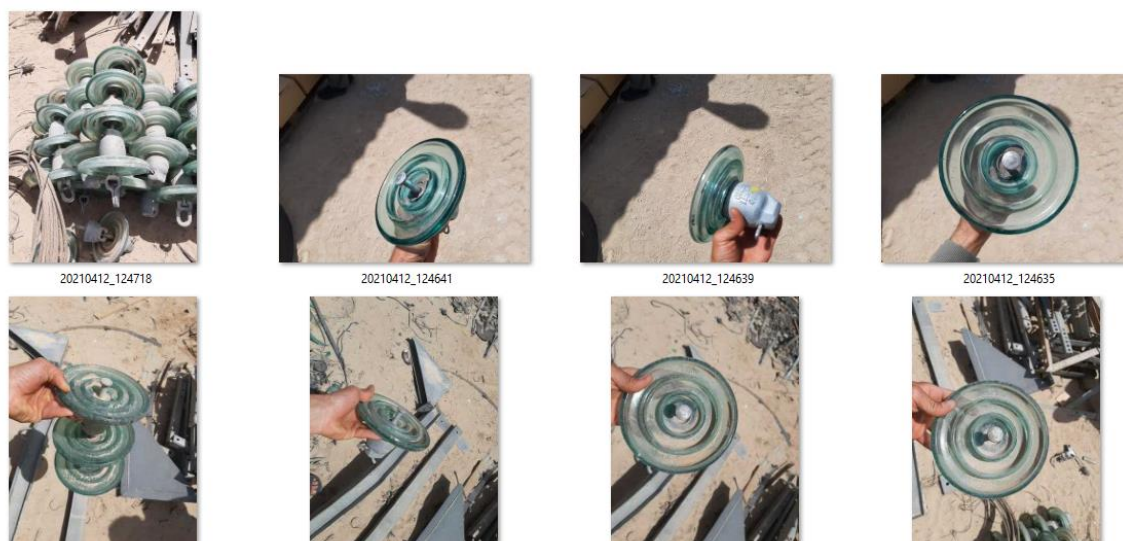


Figure III.6. Assemblage des images

Nous avons utilisés 385 images pour entraîner notre détecteur d'isolateur.

Une fois que nous avons toutes les images dont nous avons besoin, on déplaçons les 20% des images assemblées dans le répertoire \ object_detection \ images \ test, et 80% dans le répertoire \object_detection\images\train. Cela, pour assurer qu'il existe une variété d'images dans les répertoires \ test et \ train.

Tableau III.3. Classification des images assemblées

Classer Dossier	Isolateur propre	Isolateur pollué
Test	42 images	43 images
Train	150 images	150 images

III.4.5.2 Étiqueter les images :

Une fois toutes les images assemblées, On doit étiqueter les objets souhaités dans chaque image par le package *LabelImg* qui'est un excellent outil pour,

Nous téléchargeons et puis nous installons cette outil (LabelImg) pour étiqueter les images existées dans les répertoires train et test (figure III.6-7).

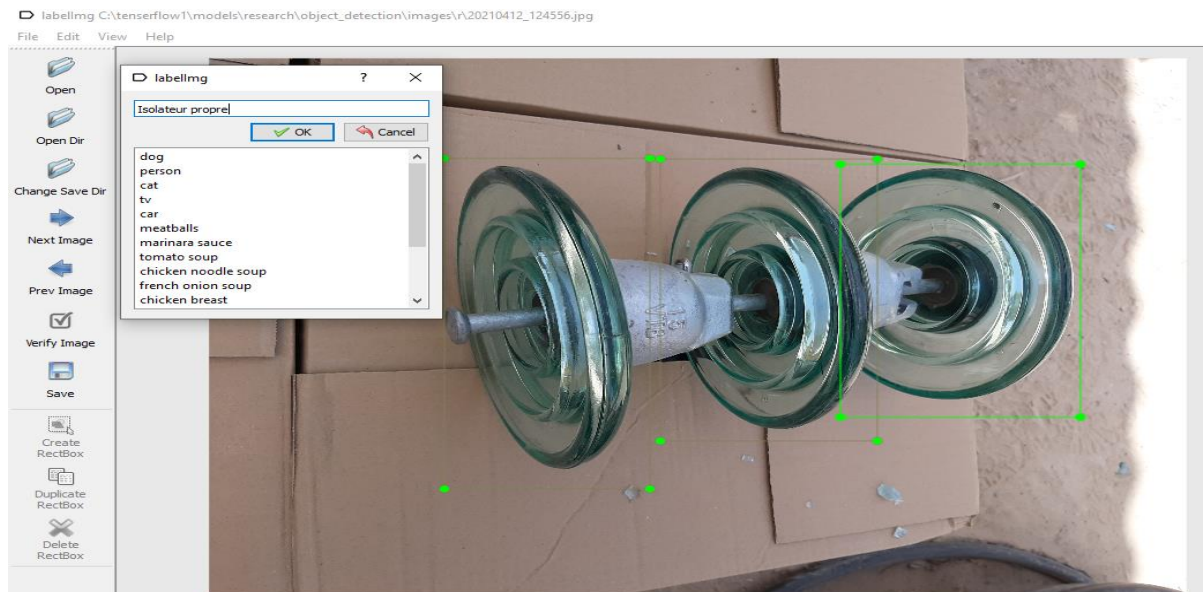


Figure III.7. Isolateur propre

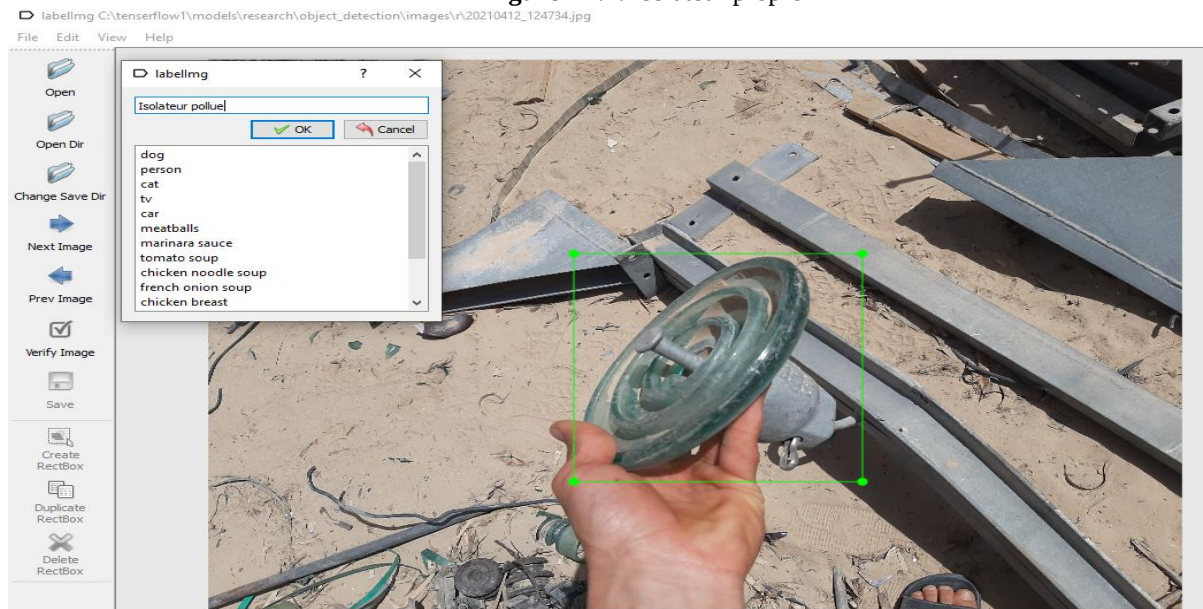


Figure III.8. Isolateur pollué

LabelImg enregistre un fichier .xml contenant les données d'étiquette pour chaque image. Ces fichiers .xml seront utilisés pour générer des TF Records, qui sont l'une des entrées du formateur TensorFlow. Une fois que nous avons étiqueté et enregistré chaque image, il y aura un fichier .xml pour chaque image dans les répertoires test et train (figure III.8).



Figure III.9. Enregistre un fichier .xml

III.4.6. Génération des données d'entraînement :

Avec les images étiquetées, on peut générer les TF Records qui servent à donner les entrées du modèle d'entraînement TensorFlow. Il faut d'abord utiliser un convertisseur en langage python `xml_to_csv.py` qui permet de convertir les images de train et les images de test en extension xml à une extension csv pour les exploiter après par le package TF Record .

La commande utilisée pour la conversion d'extension est :

(projet) `C:\tensorflow1\models\research\object_detection> python xml_to_csv.py`

```

Anaconda Prompt (anaconda3)
(base) C:\Users\user>conda activate tensorflow2
(tensorflow2) C:\Users\user>cd C:\tensorflow2\models\research\object_detection
(tensorflow2) C:\tensorflow2\models\research\object_detection>python xml_to_csv.py
Successfully converted xml to csv.
Successfully converted xml to csv.
(tensorflow2) C:\tensorflow2\models\research\object_detection>

```

Figure III.10. Création des fichier csv

Après l'exécution de convertisseur nous obtenons les fichiers `train_labels.csv` et `test_labels.csv` dans le dossier `\object_detection\ images`.

images < object_detection < research < models < tensorflow2 < القرص المحلي (C:) < هذا الكمبيوتر الشخصي

الاسم	تاريخ التعديل	النوع	الحجم
test	04/27/2021 05:55 م	مجلد ملفات	
train	04/27/2021 05:56 م	مجلد ملفات	
test_labels	05/11/2021 07:08 م	Microsoft Excel C...	15 كيلوبايت
train_labels	05/11/2021 07:08 م	Microsoft Excel C...	41 كيلوبايت

Figure III.11. Les fichier csv créés.

III.4.7. Génération et numérotations des objets :

Ensuite, nous ouvrons le fichier generate_tfrecord.py dans un éditeur de texte. On désigne l'état du surface d'isolateur pollué et propre dans le code de génération, car chaque objet est numéroté pour leur identification.

Les objets dans notre travail sont les isolateurs propres et les isolateurs pollués, ces nombres obtenus sont exploités dans l'entraînement du modèle. Nous avons donné le numéro 1 pour les isolateurs propres et le numéro 2 pour isolateurs pollués.

```
generate_tfrecord.py - C:\tensorflow2\models\research\object_detection\generate_tfrecord.p...
File Edit Format Run Options Window Help

"""
Usage:
# From tensorflow/models/
# Create train data:
python generate_tfrecord.py --csv_input=images/train_labels.csv --image_dir=im

# Create test data:
python generate_tfrecord.py --csv_input=images/test_labels.csv --image_dir=im
"""
from __future__ import division
from __future__ import print_function
from __future__ import absolute_import

import os
import io
import pandas as pd
import tensorflow as tf

from PIL import Image
from object_detection.utils import dataset_util
from collections import namedtuple, OrderedDict

flags = tf.app.flags
flags.DEFINE_string('csv_input', '', 'Path to the CSV input')
flags.DEFINE_string('image_dir', '', 'Path to the image directory')
flags.DEFINE_string('output_path', '', 'Path to output TFRecord')
FLAGS = flags.FLAGS

# TO-DO replace this with label map
def class_text_to_int(row_label):
    if row_label == 'Isolateur propre':
        return 1
    elif row_label == 'Isolateur pollue':
        return 2
    else:
        return 0
```

Figure III.12. Génération des numérotations aux objets

Nous allons créer des fichiers TF Record par l'exécution du code programme generate_tfrecord.py dans l'Anaconda

le code pour les fichiers d'entrainement s'exécute par la commande suivante :

```
python generate_tfrecord.py --csv_input=images\train_labels.csv --image_dir=images\train --output_path=train.record
```

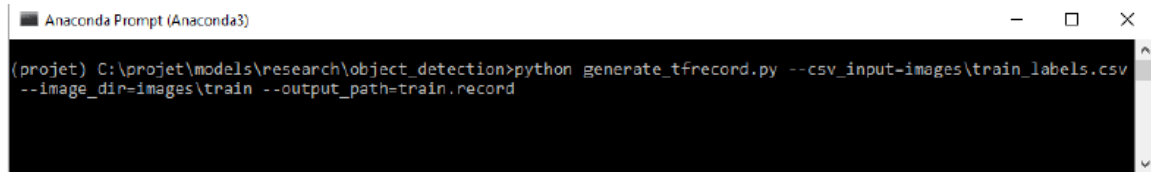


Figure III.13. Créer des fichiers TF Record de train

Après cela, nous exécutons le code des fichiers de teste par la commande suivante :

```
python generate_tfrecord.py --csv_input=images\test_labels.csv --image_dir=images\test --output_path=test.record
```

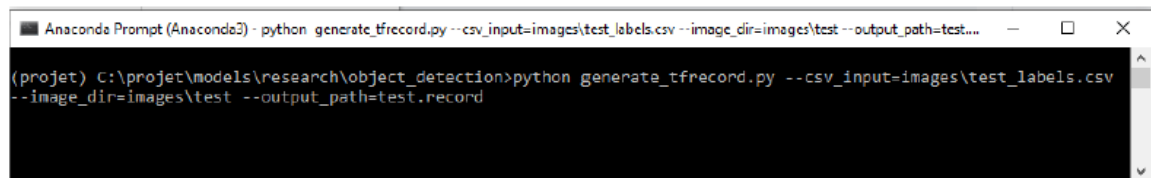


Figure III.14. Créer des fichiers de TF Record de test

Ceux-ci génèrent un fichier train.record et un fichier test.record dans le répertoire \object_detection. Ceux-ci seront utilisés pour former le nouveau classificateur de détection d'objets.

object_detection < research < models < tensorflow2 < (C:) القرص المحلي < الكمبيوتر الشخصي			
الاسم	تاريخ التعديل	النوع	الحجم
xml_to_csv	12/14/2020 06:59 م	Python File	2 كيلوبايت
<u>train.record</u>	04/29/2021 09:47 ص	ملف RECORD	67,980...
train	04/29/2021 10:21 ص	Python File	7 كيلوبايت
test3	08/04/2020 11:39 ص	ملف JPG	150 كيلوبايت
test2	08/04/2020 11:45 ص	ملف JPG	148 كيلوبايت
test1	08/07/2020 10:45 ص	ملف PNG	237 كيلوبايت
<u>test.record</u>	04/29/2021 09:48 ص	ملف RECORD	8,551...

Figure III.15. Assurez-vous créer TF Record de test et train

III.4.8. Création de la carte d'étiquettes et configuration de l'entrainement

III.4.8.1. Carte d'étiquettes

La dernière chose à faire avant l'entraînement est de créer une carte d'étiquettes et de modifier le fichier de configuration de l'entraînement.

Le mappage d'étiquettes identifie chaque objet au formateur, en définissant un mappage des noms de classe aux numéros d'identification de classe. L'utilisation d'un éditeur de texte pour créer un nouveau fichier et enregistrez-le sous labelmap.pbtxt dans le dossier C: \ tensorflow1 \ models \ research \ object_detection \ training. Dans l'éditeur de texte.

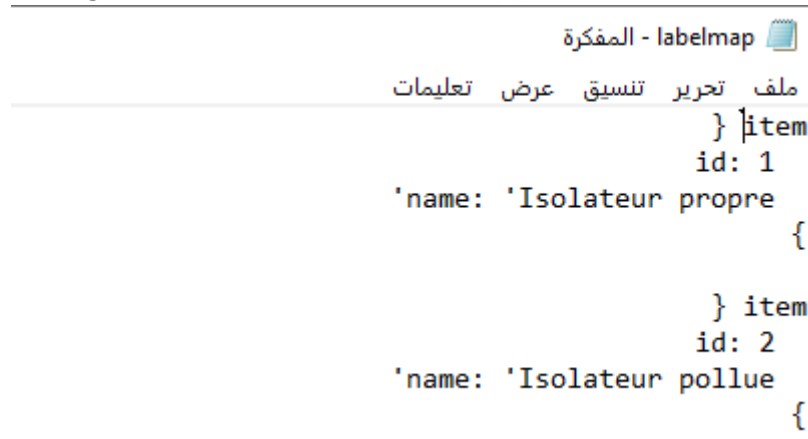


Figure III.16. Fichier labelmap.pbtxt

III.4.8.2. Configuration de l'entraînement

Enfin, le pipeline de formation à la détection d'objets doit être configuré. Il définit quel modèle et quels paramètres seront utilisés pour la formation. C'est la dernière étape avant de courir l'entraînement.

1-classement d'objet (Figure III.17.) en deux différentes classes, dans notre travail il y'a deux classes des objets isolateur propre et isolateur pollue num_classes : 2.

```

} model
} faster_rcnn
num_classes: 2
} image_resizer
} keep_aspect_ratio_resizer
min_dimension: 600
max_dimension: 1024

```

Figure III.17. Classification des fichiers

2- Configuration du fichier **fine_tune_checkpoint** dans le dossier d'entraînement

3- Configuration des fichiers **input_path** et **label_map_path**

4- Configuration de l'ensemble de données d'évaluation par le fichier

```

} :train_input_reader
} tf_record_input_reader
input_path:
"C:/tensorflow2/models/research/object_detection/train.record
{
label_map_path:
"C:/tensorflow2/models/research/object_detection/training/labelm
"ap.pbtxt

```

Figure III.118. Classeur train_input_reader

5-Désignation de l'emplacement des données de validation par le classeur `eval_input_reader` :

```

} :eval_config
"metrics_set": "coco_detection_metrics
num_examples: 84
{

} :eval_input_reader
} tf_record_input_reader
input_path:
""C:/tensorflow/models/research/object_detection/test.record
{
label_map_path: "
C:/tensorflow2/models/research/object_detection/training/labelma
" p.pbtxt
shuffle: false

```

Figure III.19. Classeur `eval_input_reader`

Enregistrez le fichier une fois les modifications apportées.

III.4.9. Organisation de l'entraînement

Maintenant, il va commencer un vrai travail. L'ordinateur va apprendre de l'ensemble de données et créer un réseau neuronal. Comme nous modélisons l'entraînement sur une version CPU (processeur), cela prendra plusieurs jours pour obtenir un bon résultat.

L'exécutons de l'entraînement ce fait par la commande suivant dans l'Anaconda :

`python train.py --logtostderr --train_dir=training/ --
pipeline_config_path=training/faster_rcnn_inception_v2_pets.config`

```

C:\WINDOWS\system32\cmd.exe - python train.py --logtostderr --train_dir=training/ --pipeline_config_path=training/faster_rcnn_inception_v2_pets.c...
[device.cc:1195] Creating TensorFlow device (/device:GPU:0) -> (device: 0, name: GeForce GTX 1060 6GB, pci bus id: 0000:01:00.0, compute capability: 6.1)
INFO:tensorflow:Restoring parameters from C:/tensorflow1/models/research/object_detection/faster_rcnn_inception_v2_coco_2018_01_28/model.ckpt
INFO:tensorflow:Starting Session.
INFO:tensorflow:Saving checkpoint to path training/model.ckpt
INFO:tensorflow:Starting Queues.
INFO:tensorflow:global_step/sec: 0
INFO:tensorflow:Recording summary at step 0.
INFO:tensorflow:global step 1: loss = 2.6708 (5.383 sec/step)
INFO:tensorflow:global step 2: loss = 3.0352 (0.251 sec/step)
INFO:tensorflow:global step 3: loss = 3.4884 (0.204 sec/step)
INFO:tensorflow:global step 4: loss = 2.9733 (0.193 sec/step)
INFO:tensorflow:global step 5: loss = 2.2184 (0.191 sec/step)
INFO:tensorflow:global step 6: loss = 2.0321 (0.554 sec/step)
INFO:tensorflow:global step 7: loss = 2.0424 (0.211 sec/step)
INFO:tensorflow:global step 8: loss = 2.0252 (0.208 sec/step)
INFO:tensorflow:global step 9: loss = 2.0053 (0.194 sec/step)
INFO:tensorflow:global step 10: loss = 1.3622 (0.193 sec/step)
INFO:tensorflow:global step 11: loss = 1.8027 (0.197 sec/step)
INFO:tensorflow:global step 12: loss = 1.2485 (0.196 sec/step)
INFO:tensorflow:global step 13: loss = 1.0712 (0.193 sec/step)
INFO:tensorflow:global step 14: loss = 1.6604 (0.189 sec/step)
INFO:tensorflow:global step 15: loss = 1.2657 (0.192 sec/step)
INFO:tensorflow:global step 16: loss = 1.4351 (0.193 sec/step)
INFO:tensorflow:global step 17: loss = 1.2152 (0.192 sec/step)
INFO:tensorflow:global step 18: loss = 1.1165 (0.197 sec/step)
INFO:tensorflow:global step 19: loss = 1.6557 (0.192 sec/step)
INFO:tensorflow:global step 20: loss = 1.7777 (0.200 sec/step)

```

Figure III.20. Exécution de l'entraînement

Nous pouvons afficher la progression de la tâche d'entraînement à l'aide de TensorBoard. Pour ce faire, il faut ouvrir une nouvelle instance d'Anaconda Prompt, activation d'environnement virtuel tensorflow1, accédez au répertoire C:\tensorflow1\models\research\object_detection et exécutez la commande suivante :

(tensorflow1) C:\tensorflow1\models\research\object_detection>tensorboard --logdir=training

```
Anaconda Prompt (anaconda3) - tensorboard --logdir=training
(base) C:\Users\user>conda activate projet
(projet) C:\Users\user>cd C:\tensorflow2\models\research\object_detection
(projet) C:\tensorflow2\models\research\object_detection>tensorboard --logdir=training
TensorBoard 1.12.2 at http://DESKTOP-NML4KF7:6006 (Press CTRL+C to quit)
```

Figure III.21. Contrôler l'évaluation d'entrainement

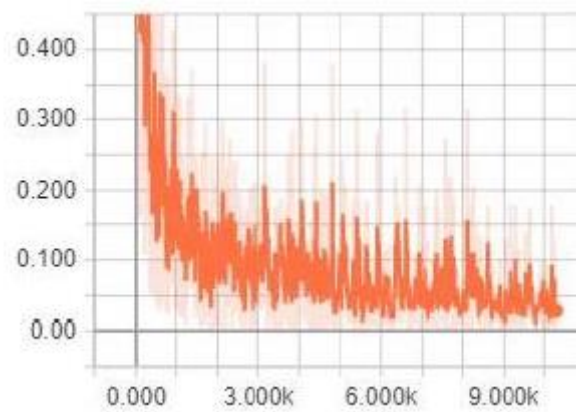
Cela va créer une page Web sur l'ordinateur local à YourPCName: 6006, qui peut être visualisée via un navigateur Web. La page TensorBoard fournit des informations et des graphiques qui montrent la progression de la formation. Un graphique des erreurs montre la perte globale du classificateur au fil du temps



(A). Step 117



(B). step 3302



(C). step 14868

Figure III.22.(A)(B)(C): Les courbes de losses.

Après 14868 pas ou la perte égale à 0,0089 qu'il devient suffisante pour que le modèle puisse produire des résultats précis, maintenant, nous pouvons arrêter l'entraînement.

La routine d'entraînement enregistre périodiquement des points de contrôle toutes les cinq minutes .

training < object_detection < research < models < tensorflow2 < (C:) القرص المحلي < الكمبيوتر الشخصي				
الاسم	تاريخ التعديل	النوع	الحجم	
checkpoint	05/04/2021 10:34 م	ملف	1 كيلوبايت	
events.out.tfevents.1619688613.DESKTOP...	05/04/2021 10:38 م	ملف	2,069,853 ...	
faster_rcnn_inception_v2_pets	04/29/2021 10:26 ص	ملف	4 كيلوبايت	
graph	04/29/2021 10:29 ص	ملف	10,108... كيلو	
labelmap	04/29/2021 09:57 ص	ملف	1 كيلوبايت	
model.ckpt-14788.data-00000-of-00001	05/04/2021 09:50 م	ملف	100,596...	
model.ckpt-14788.index	05/04/2021 09:50 م	ملف	25 كيلوبايت	
model.ckpt-14788.meta	05/04/2021 09:53 م	ملف	5,255...	
model.ckpt-14813.data-00000-of-00001	05/04/2021 10:00 م	ملف	100,596...	
model.ckpt-14813.index	05/04/2021 10:01 م	ملف	25 كيلوبايت	
model.ckpt-14813.meta	05/04/2021 10:06 م	ملف	5,255...	
model.ckpt-14828.data-00000-of-00001	05/04/2021 10:10 م	ملف	100,596...	
model.ckpt-14828.index	05/04/2021 10:10 م	ملف	25 كيلوبايت	
model.ckpt-14828.meta	05/04/2021 10:12 م	ملف	5,255...	
model.ckpt-14857.data-00000-of-00001	05/04/2021 10:20 م	ملف	100,596...	
model.ckpt-14857.index	05/04/2021 10:20 م	ملف	25 كيلوبايت	
model.ckpt-14857.meta	05/04/2021 10:31 م	ملف	5,255...	
model.ckpt-14868.data-00000-of-00001	05/04/2021 10:34 م	ملف	100,596...	
model.ckpt-14868.index	05/04/2021 10:34 م	ملف	25 كيلوبايت	
model.ckpt-14868.meta	05/04/2021 10:38 م	ملف	5,255...	
pipeline	04/29/2021 10:27 ص	ملف	4 كيلوبايت	

Figure III.23. Dossiers d'entrainement

III.4.10. Exporter le graphiques

Maintenant que l'entraînement est terminé, la dernière étape consiste à générer les graphes d'inférence (fichier. pb). Depuis le dossier \ object_detection,

```
Anaconda Prompt (anaconda3) - python export_inference_graph.py --input_type image_tensor --pipeline_config_path training/faster_rcnn_inception_v...
(base) C:\Users\user>conda activate projet
(projet) C:\Users\user>cd C:\tensorflow2\models\research\object_detection
(projet) C:\tensorflow2\models\research\object_detection>python export_inference_graph.py --input_type image_tensor --pi
pipeline_config_path training/faster_rcnn_inception_v2_pets.config --trained_checkpoint_prefix training/model.ckpt-XXXX --
output_directory inference_graph
```

Figure III.24. Création du graphe d'inférence

Cela se crée un fichier Frozen_inference_graph.pb dans le dossier \ object_detection \ inference_graph. Le fichier contient le classificateur de détection d'objet

inference_graph < object_detection < research < models < tensorflow2 < (C:) القرص المحلي < الكمبيوتر الشخصي

الاسم	تاريخ التعديل	النوع	الحجم
saved_model	05/08/2021 05:32 م	مجلد ملفات	
checkpoint	05/06/2021 05:19 م	ملف	1 كيلوبايت
frozen_inference_graph.pb	05/06/2021 05:20 م	ملف PB	50,952... كيلو
model.ckpt.data-00000-of-00001	05/06/2021 05:19 م	ملف DATA-00000	50,337... كيلو
model.ckpt.index	05/06/2021 05:19 م	ملف INDEX	16 كيلوبايت
model.ckpt.meta	05/06/2021 05:20 م	ملف META	1,616... كيلوبايت
pipeline	05/05/2021 06:11 م	ملف CONFIG	4 كيلوبايت

Figure III.25. Créé un fichier Frozen_inference_graph.pb

III.4.11. Test et diagnostique de l'état de surface des isolateurs de haute tension par le modèle réalisé:

Le classeur de détection d'objet est complètement prêt, pour tester une image, une vidéo ou une webcam on écrit des scripts Python.

Avant d'exécuter les scripts Python, il faut modifier la variable NUM_CLASSES dans le script pour qu'elle corresponde au nombre de classes que vous souhaitez découvrir. (Pour mon détecteur diélectrique, j'ai besoin de deux NUM_CLASSES

Pour tester votre détecteur d'objet, déplacez une image de l'objet ou des objets dans le dossier \ object_detection et modifiez la variable IMAGE_NAME dans Object_detection_image.py pour qu'elle corresponde au nom de fichier de l'image. Vous pouvez également utiliser une vidéo des objets (en utilisant Object_detection_video.py), ou simplement brancher une webcam USB et la pointer vers les objets (en utilisant Object_detection_webcam.py).

- 1-Activation de l'environnement du notre projet dans l'espace Anaconda.
- 2- Exécution de la commande « idle » pour ouvrir le python.



Figure III.26. Ouverture de l'interface python

Ensuite, l'interface python s'ouvre et nous exécutons notre modèle **object_detection_image.py** [49] et nous déterminons les différents paramètres suivants :

- 1) **NUM_CLASSES** c'est le nombre d'objets dans notre modèle, qui sont deux objets décrivent l'état de surfaces des isolateurs (isolateur propre et isolateur pollue), alors **NUM_CLASSES=2**
- 2) **MODEL_NAME** = inference_graph.
- 3) **IMAGE_NAME** c'est les images de test; **IMAGE_NAME=test.jpg**

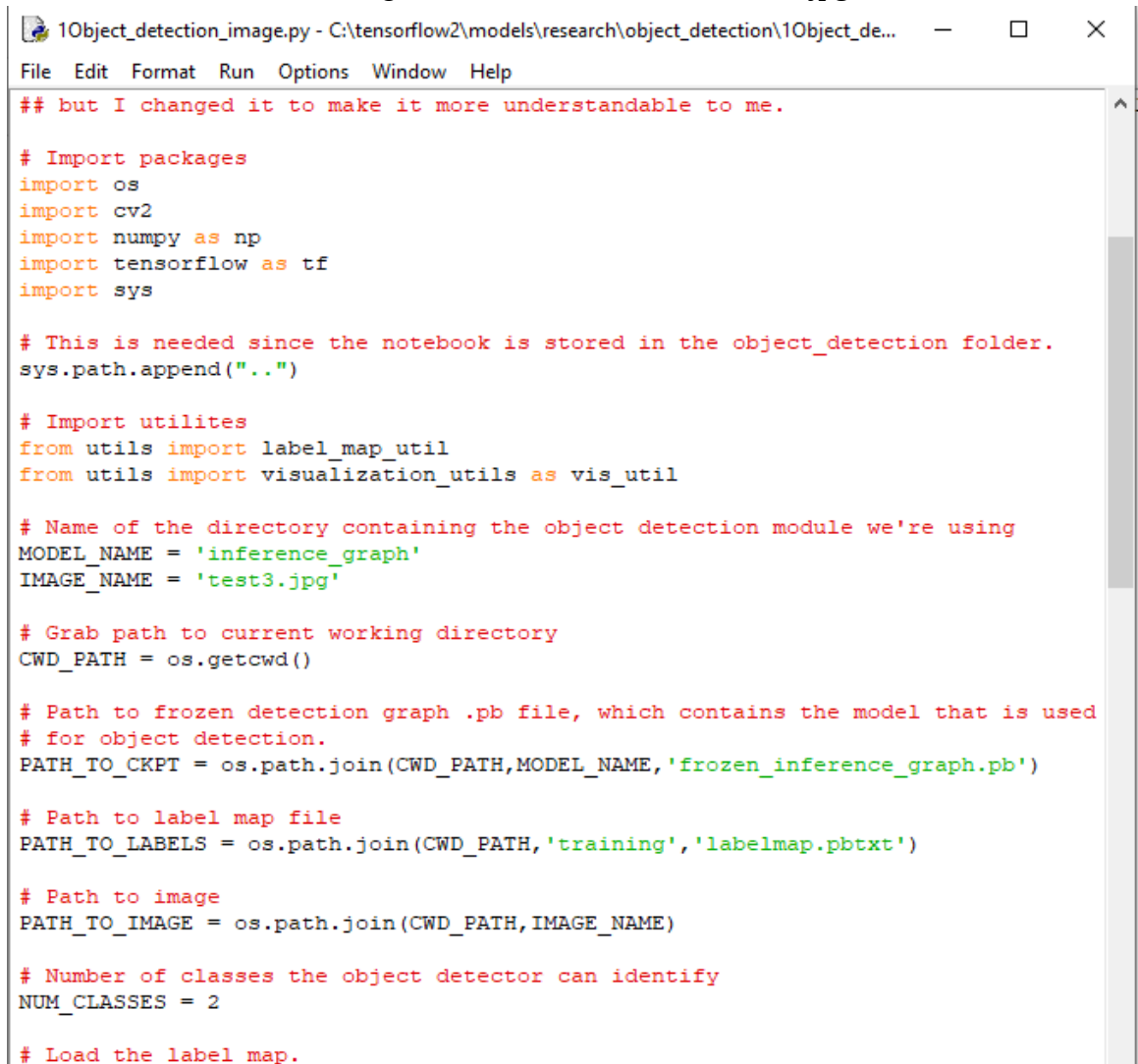


Figure III.27. Configuration du modèle de détection de l'état de surface des isolateurs

Après la configuration des paramètres du modèle, nous exécutons le fichier python de la détection de l'état de la surface des isolateurs.

Aussi ; peut utiliser `object_detection_video.py` ou `object_detection_webcam.py`

III.5. Présentation des résultats de l'état de surface des isolateurs :



Figure III.28. Isolateur propre



Figure III.29. Isolateur pollutes

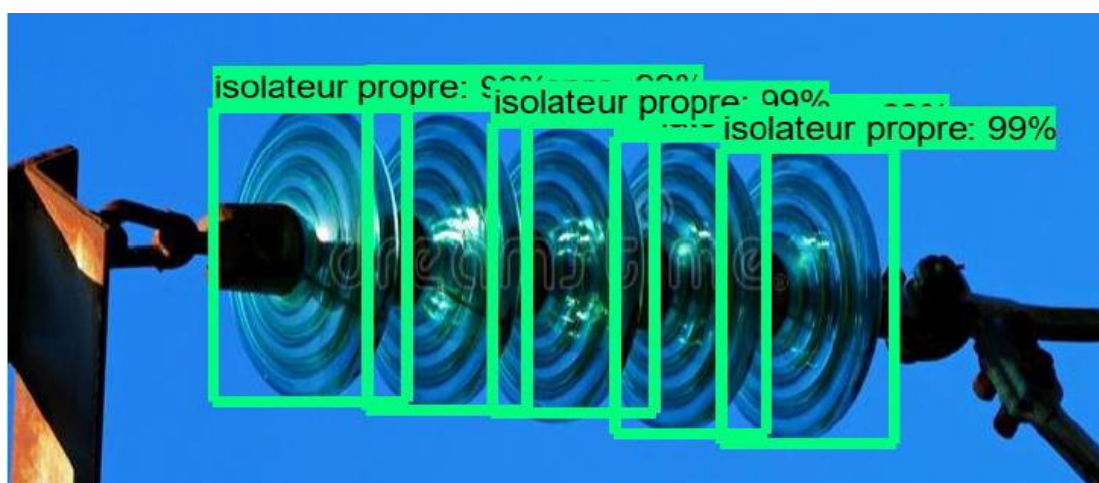


Figure III.30. Chaîne isolateurs propres

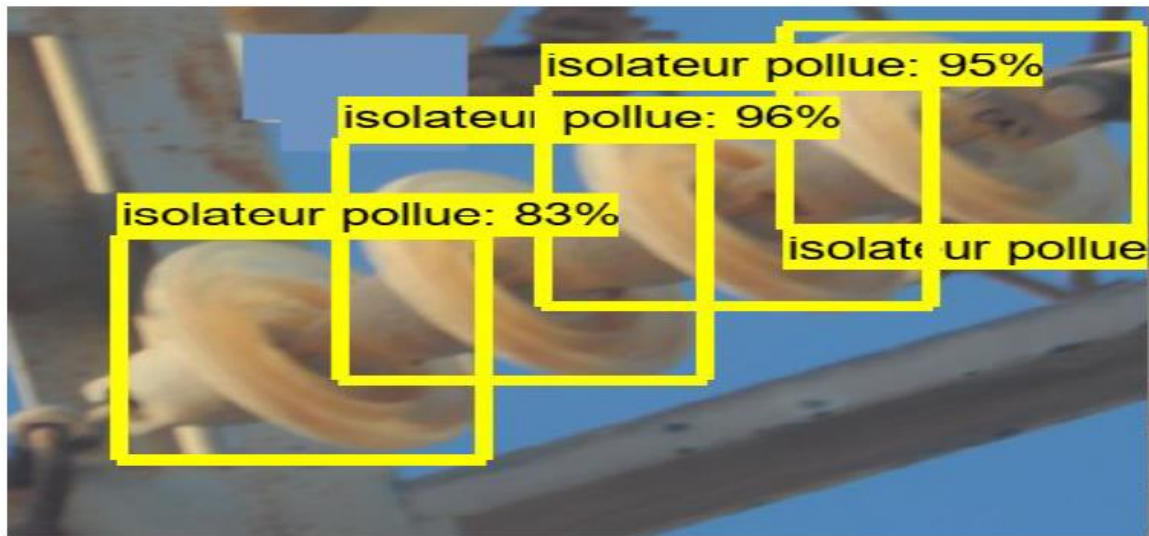


Figure III.31. Chaîne isolateurs pollués

III.6. Conclusion :

Nous avons réussi à mettre au point un modèle de reconnaissance d'isolateurs via des images statiques, une image capturée par un web Cam et enfin une image défilante sur un vidéo.

Ce modèle permet de faire détecter l'état d'isolateur est ce qu'il est propre ou pollué. Notre modèle a été entraîné à travers des techniques d'intelligence artificielle incorporé dans la bibliothèque TensorFlow programmé par le langage python.

Les résultats obtenus démontrent que notre modèle est a réussi à détecter l'état d'isolateur dans la chaîne et en solo.

Chapitre IV:

La détection d'état de surface d'isolateur par une carte Raspberry Pi

IV.1. Introduction :

Un système de reconnaissance d'objets est une technologie capable de faire correspondre des objets à partir d'images numériques ou d'images vidéo à des bases de données d'objets. Bien que les humains puissent reconnaître des objets sans trop d'effort, la reconnaissance d'objets présente un problème difficile pour la reconnaissance de formes en informatique. Le système de reconnaissance d'objets cherche à reconnaître l'état de la surface de l'isolant du réseau électrique en 3D et modifie son apparence en éclairant et en exprimant l'isolant à partir de son image 3D. Pour accomplir cette tâche de calcul, le système de reconnaissance d'objets exécute quatre étapes. La première détection de tampon est utilisée pour séparer le tampon de l'arrière-plan de l'image. Dans une deuxième étape, les images d'isolateur segmentées sont ajustées pour prendre en compte les positions d'isolateur, les tailles d'image et les caractéristiques d'imagerie, telles que l'éclairage et les niveaux de gris. Le but du processus d'alignement est de permettre une localisation correcte des caractéristiques de l'isolateur dans la troisième étape, qui consiste à extraire les profils d'isolateur. Des éléments tels que le verre et la saleté sont affichés et mis à l'échelle dans les images pour représenter l'isolant. Le vecteur a un visage fort, puis dans la quatrième étape, il est comparé à la base de données tampon.

IV.2. Liste de matériel nécessaire :

a) Raspberry Pi 3 model b:

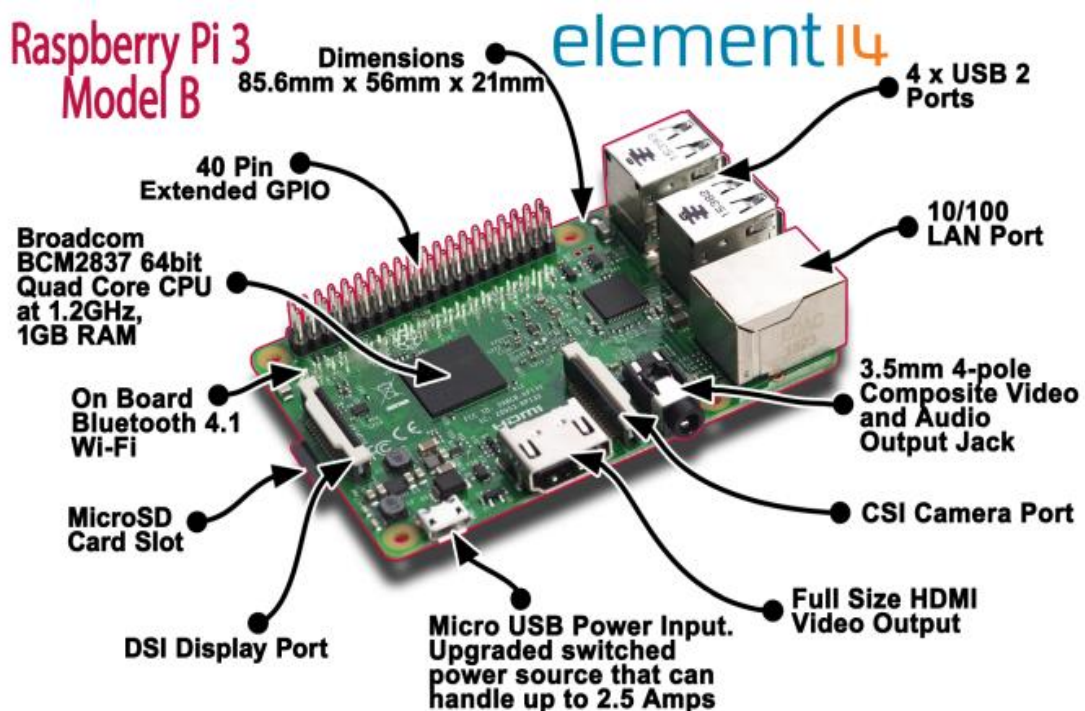


Figure IV.1. Raspberry Pi 3 model b

Pour mettre en œuvre un tel projet, l'étape principale et la plus importante était de trouver le matériel à utiliser pour la détection d'état de surface des isolateurs. Nous avons

choisi un Raspberry Pi modèle B3 à utiliser dans notre travail. Nous avons fait beaucoup de recherches et comparé des éléments dans différents microcontrôleurs, comme, le coût, le traitement et la convivialité. Les principales raisons pour lesquelles nous avons choisi ces éléments spécifiques sont la capacité des traitements élevés, le prix relativement bas et sa capacité à s'adapter à différentes programmations modes. L'appareil utilise Linux comme système d'exploitation, qui a accès à un grand nombre de bibliothèques et d'applications compatible avec celui-ci. Raspberry Pi a un port Ethernet nous permettant une connexion réseau, tant que nous sommes dans le même sous-réseau avec l'appareil auquel nous voulons accéder et gérer, 4 ports USB utilisés pour connecter des appareils comme un clavier, une souris, appareil photo, et d'autres appareils qui se connectent via un port USB, et un port HDMI nous donnant accès à l'interface du système d'exploitation installé et peut également être utilisé la première fois lors de l'installation des appareils. Il dispose de 40 broches qui nous permettent de recevoir et d'envoyer des signaux. Ils sont divisés en deux en deux groupes : le groupe 3V et le groupe 5V. Par conséquent, un côté du microcontrôleur donne une tension de 3V, et l'autre 5V. Outre les 40 broches de tension, il a des broches qui sont utilisées pour recevoir signaux, qui dans notre cas a été utilisé pour connecter le bouton, qui enverra le signal pour l'identification du visage. Framboise Pi n'a pas de système d'exploitation préalablement installé, mais qui peut être téléchargé à partir du site Web de Raspberry, et transféré sur une carte SD, la figure 1 montre le modèle Raspberry pi B 3 avec ses composants La fondation fournit les distributions ARM Debian et Arch Linux ainsi que Python comme programmation principale langage, avec le support de BBC BASIC, C et Perl. Le système est programmé en utilisant la programmation Python Langue. Nous avons développé des algorithmes pour la détection et la reconnaissance des visages pour la sécurité.

b) Camera USB :

Une webcam, ou une caméra électronique, est une caméra conçue pour être utilisée comme périphérique informatique et qui produit des vidéos et des images qui ne sont pas destinées à atteindre une haute qualité, destinées à capturer les images et à détecter des objets.



Figure IV.2. Camera USB

c) microSD card 32GB :

Une carte mémoire afin d'y charger le système d'exploitation. Cette carte doit faire plus de 8 Go car le système d'exploitation prend un grand espace de stockage. Il doit également y avoir de l'espace pour les bagages et le travail requis.



Figure IV.3. microSD card 32GB

d) Alimentation pour Raspberry pi3:

Pour le modèle B du Raspberry Pi 3, vous allez avoir besoin d'une alimentation avec un courant de sortie maximum à 2.5 A et une tension de 5V.



Figure IV.4. Alimentation de Raspberry pi

e) Un câble HDMI :

Le câble HDMI servira simplement pour un écran, ce qui sera quand même très pratique pour installer l'OS et utiliser notre Pi, d'autant plus que c'est nouveau pour nous .



Figure IV.5. câble HDMI

f) Un clavier et une souris :

Un clavier et une souris sera obligatoire pour installer le système d'exploitation et paramétrer votre Raspberry Pi



Figure IV.6. Clavier et souris

g) Ventilateur :

Les ventilateurs axiaux sont utilisés dans toutes les industries où le refroidissement des composants ou des machines est important. Les ventilateurs fonctionnent en aspirant l'air dans une direction axiale à travers une surface pour la refroidir ou en repoussant l'air chaud pour la refroidir. Ceci est essentiel dans les objets de tous les jours tels qu'un ordinateur de bureau, où le ventilateur axial refroidirait le processeur pour garantir que l'ordinateur continuerait à fonctionner correctement. Les problèmes de surchauffe s'appliquent à la plupart des applications électroniques ou d'ingénierie, et les ventilateurs axiaux sont donc couramment utilisés.



Figure IV.7. Ventilateur

IV.3. Modules et toolbox nécessaires :**IV.3.1. Traitement d'image avec open cv :**

OpenCV (pour Open Computer Vision) est une bibliothèque graphique libre, initialement développée par Intel, spécialisée dans le traitement d'images en temps réel.

Toutes les classes et les méthodes d'OpenCV en C++ ont un espace de travail (name space) nommé cv. Ainsi on doit utiliser la directive de compilation suivante après avoir déclaré les fichiers d'en tête en utilisant l'espace de noms cv.

Une image peut être transmise à l'intérieur d'une structure en C du type `CvMat` ou `IplImage`. Ces structures sont issues des versions 1.x de Open cv. La structure `IplImage` est un vieux format d'image original compatible Intel IPP.



Figure IV.8. L'insigne de opencv

IV.3.2. Méthodes de l'objet réognition :

La reconnaissance d'objets est un problème difficile en analyse d'images et en vision par ordinateur. La sécurité de l'information est devenue très importante et difficile. Des caméras de sécurité sont actuellement déployées dans tous les emplacements dotés d'un système de sécurité. La reconnaissance d'objets est un système biométrique utilisé pour identifier ou vérifier un objet à travers une image numérique. Le système de reconnaissance d'objets est utilisé en sécurité. Le système de reconnaissance d'objet doit être capable de détecter automatiquement l'objet dans l'image. Il s'agit d'extraire leurs caractéristiques puis de les reconnaître, en dehors de l'illumination, de l'expression, de l'illumination, du vieillissement et des transitions (traduction, rotation et mise à l'échelle de l'image), ce qui est une tâche difficile.

La première section décrit les méthodes courantes telles que la méthode de correspondance globale, la méthode d'extraction de caractéristiques et les méthodes mixtes. La deuxième section décrit des applications avec des exemples et la troisième section décrit enfin les futures directions de recherche pour l'apprentissage d'objets.

IV.3.3. Imutils:

Imutils est une série de fonctions pratiques pour accélérer le calcul OpenCV sur le Raspberry Pi.

Maintenant que nous savons quel Raspberry Pi choisir et les différents accessoires importants pour son fonctionnement et son utilisation, nous pouvons passer à quelque chose de plus concret, le choix du système d'exploitation.

IV.4. Installation de système d'exploitation sur Raspberry pi (Raspbian) :

Raspbian est le système d'exploitation de référence pour Raspberry Pi. Il est basé sur Linux Debian et il est très régulièrement mis à jour.

Optimisé spécialement pour Raspberry Pi, c'est une distribution polyvalente qui vous permettra de vous familiariser très facilement avec le matériel.

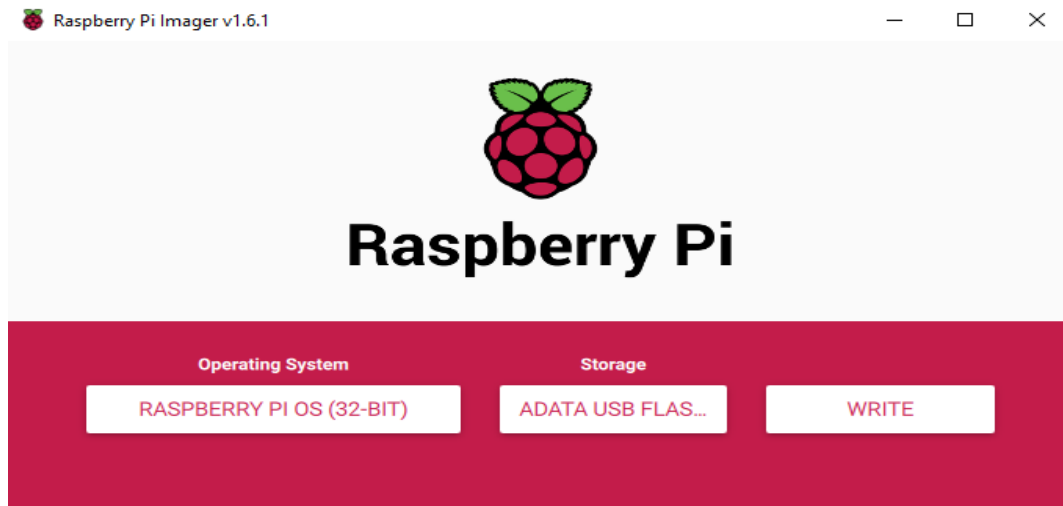


Figure IV.9. Le site officiel pour l'installation du système d'exploitation sur la carte mémoire

IV.5. Exécutez Raspberry Pi et installez des packages :

Après avoir installé tous les éléments ci-dessus. Connectez le Raspberry Pi au câble d'alimentation Ouvrez un terminal. Vous pouvez le faire en appuyant sur CTRL + T.

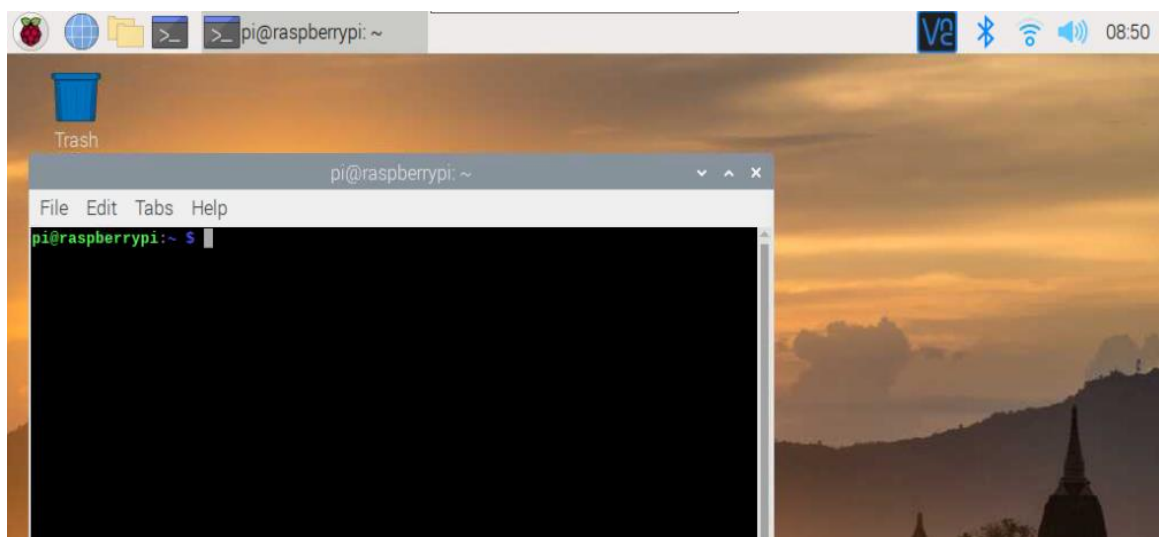


Figure IV.10. Premier fonctionnement de raspberrypi

Nous installons OpenCV, face_recognition et imutils en exécutant les commandes suivantes dans Terminal.

Tableau. IV.1. Installons OpenCV, face_recognition et imutils

Étape	Code d'installer OpenCV
1	<code>sudo apt-get purge wolfram-engine -y</code>
2	<code>sudo apt-get purge libreoffice* -y</code>
3	<code>sudo apt-get clean</code>
4	<code>sudo apt-get autoremove -y</code>
5	<code>sudo apt-get update</code>
6	<code>sudo apt-get upgrade</code>
7	<code>sudo apt-get install python-opencv</code>
8	<code>sudo apt-get install build-essential cmake pkg-config</code>
9	<code>sudo apt-get install libjpeg-dev libtiff5-dev libjasper-dev libpng12-dev</code>
10	<code>sudo apt-get install libavcodec-dev libavformat-dev libswscale-dev libv4l-dev</code>
11	<code>sudo apt-get install libxvidcore-dev libx264-dev</code>
12	<code>sudo apt-get install libgtk2.0-dev libgtk-3-dev</code>
13	<code>sudo apt-get install libatlas-base-dev gfortran</code>
14	<code>sudo apt-get install libqtgui4</code>
15	<code>sudo modprobe bcm2835-v4l2</code>
16	<code>sudo apt-get install libqt4-test</code>
17	<code>sudo apt-get install python3-dev</code>
18	<code>sudo apt-get install python3-pip</code>
19	<code>pip3 install opencv-python</code>
	Code d'installer face_recognition
1	<code>Pip3 install face-recognition</code>
	Code d'installer imutils
1	<code>Pip install impiputils</code>

L'installation de ce bagage prend beaucoup de temps, elle dépasse environ 24 heures.

IV.6. Entraîner le modèle à reconnaître des objets :

Dans cette section, nous allons nous concentrer sur la formation de notre Pi sur l'isolateur que nous voulons qu'il apprenne.

a) Création des fichiers fondamentaux :

Collectons maintenant l'ensemble des données que nous utiliserons pour entraîner le Pi. Depuis le bureau Raspberry Pi, nous ouvrons un gestionnaire de fichiers en cliquant sur l'icône du dossier.

Nous passons au dossier facial_recognition puis au dossier dataset. Cliquez avec le bouton droit dans le dossier du jeu de données et sélectionnez Nouveau dossier.

Nous entrerons le nom de la première chose pour le nom du dossier nouvellement créé.

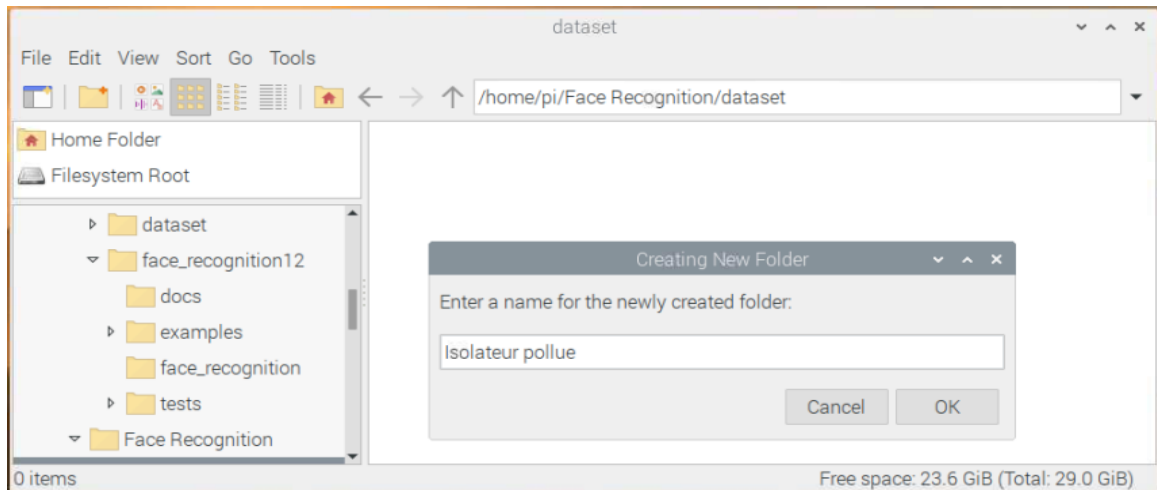


Figure IV.11. Création des fichiers fondamentaux

b) Premier code :

Commençons maintenant à écrire le premier code. Un code doit être créé dans le terminal Python IDE pour ouvrir notre caméra. Nommez ensuite l'image en fonction du nom mentionné dans le code. Ensuite, il enregistre les images dans le dossier créé en dernier

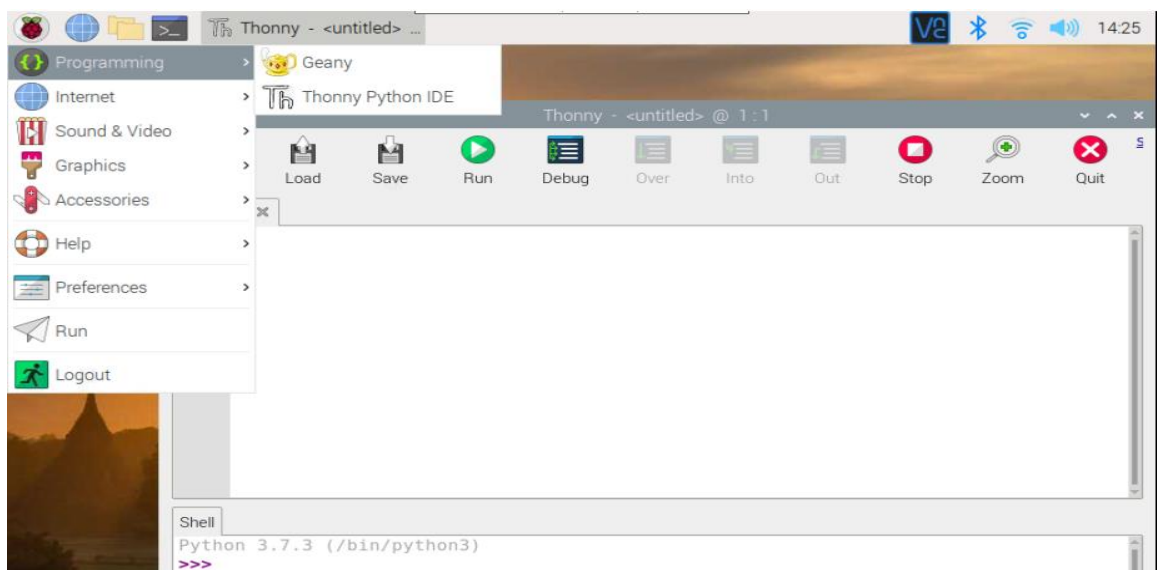
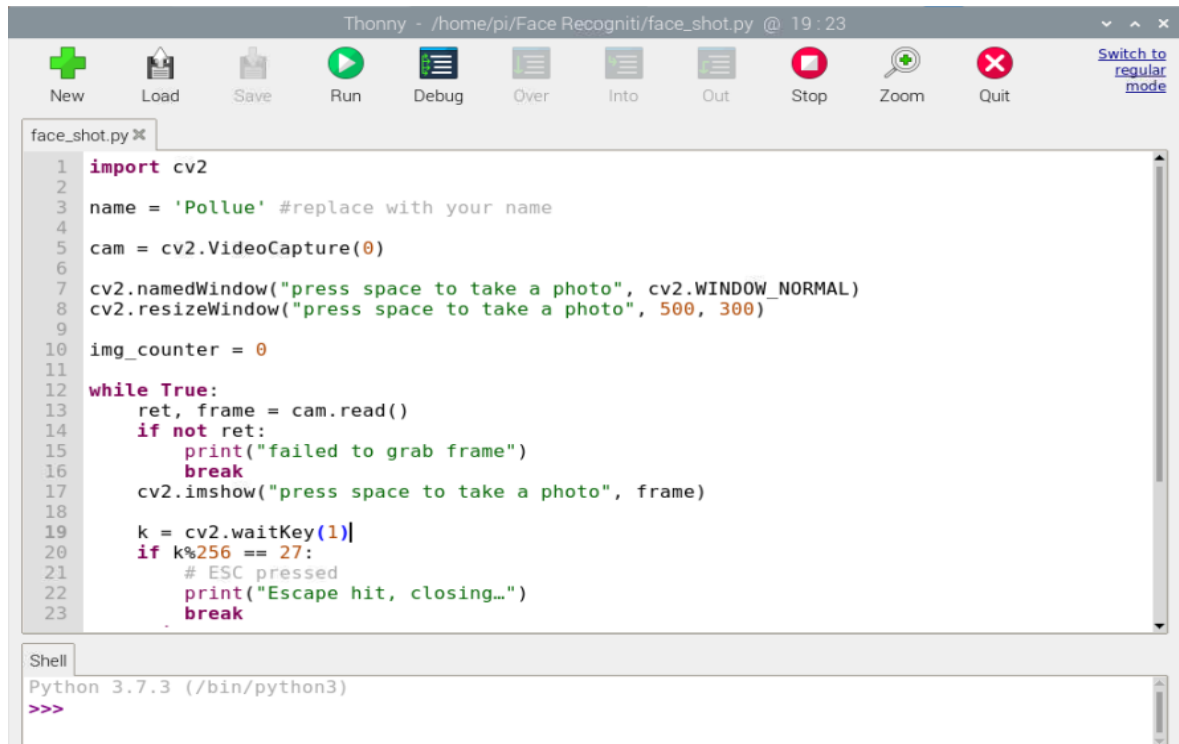


Figure IV.12. Terminal IDE du Python

Nous écrivons le code suivant :



```

1 import cv2
2
3 name = 'Pollue' #replace with your name
4
5 cam = cv2.VideoCapture(0)
6
7 cv2.namedWindow("press space to take a photo", cv2.WINDOW_NORMAL)
8 cv2.resizeWindow("press space to take a photo", 500, 300)
9
10 img_counter = 0
11
12 while True:
13     ret, frame = cam.read()
14     if not ret:
15         print("failed to grab frame")
16         break
17     cv2.imshow("press space to take a photo", frame)
18
19     k = cv2.waitKey(1)
20     if k%256 == 27:
21         # ESC pressed
22         print("Escape hit, closing...")
23         break

```

Figure IV.13. Premier code

L'appuyer sur l'icône Run dans l'IDE Python exécute le code Face_shot .py.

Une nouvelle fenêtre s'ouvrira donnant sur notre webcam. (Sur le Raspberry Pi 3, il a fallu environ 10 secondes pour que la fenêtre de visualisation de la webcam s'ouvre.)

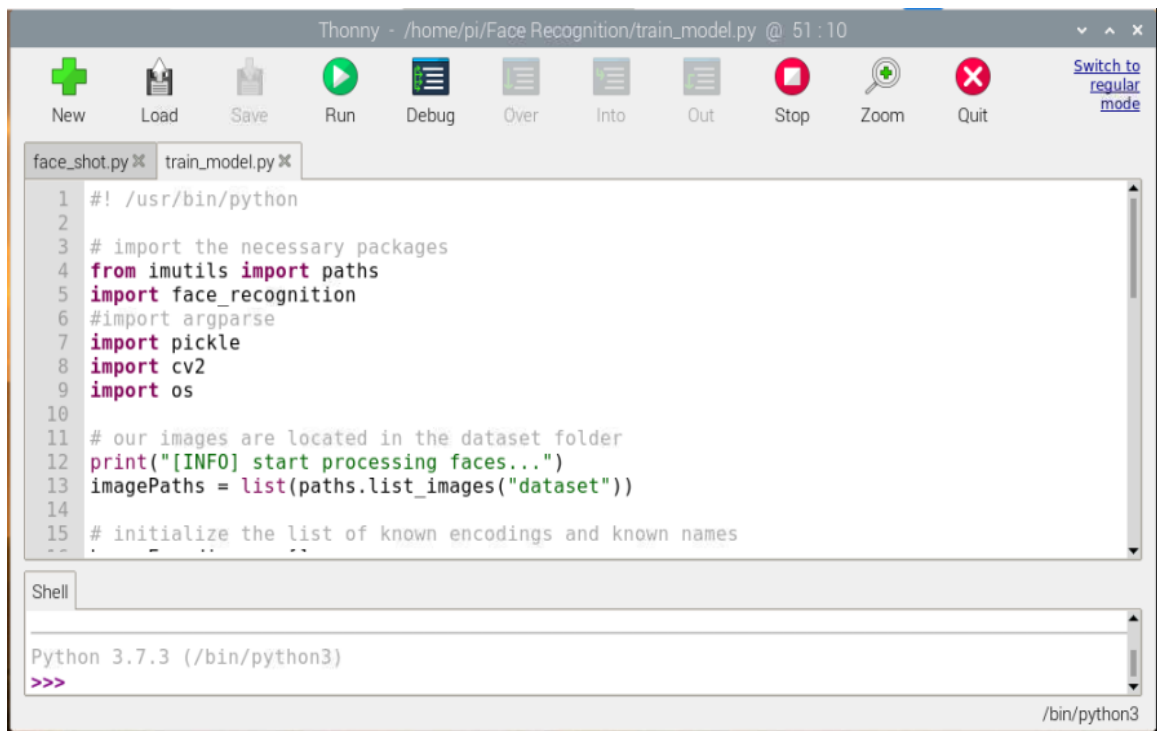
On Pointe la webcam sur l'isolateur et appuyons sur la barre espace pour prendre une photo de l'isolateur; Chaque fois que nous appuyons sur la barre d'espace, nous prenons une autre photo. Nous vous recommandons de prendre environ 10 photos en secours pour différents angles. Ces images seront utilisées pour entraîner notre modèle. Appuyer sur Echap lorsque nous terminons de prendre des photos.



Figure IV.14. Exécution du code et l'enregistrement des photos pour les isoleurs

c) Deuxième code :

Le deuxième code est spécialisé dans l'analyse des images de l'isolant et la sauvegarde des données de chaque image pour faciliter le processus de pré-reconnaissance



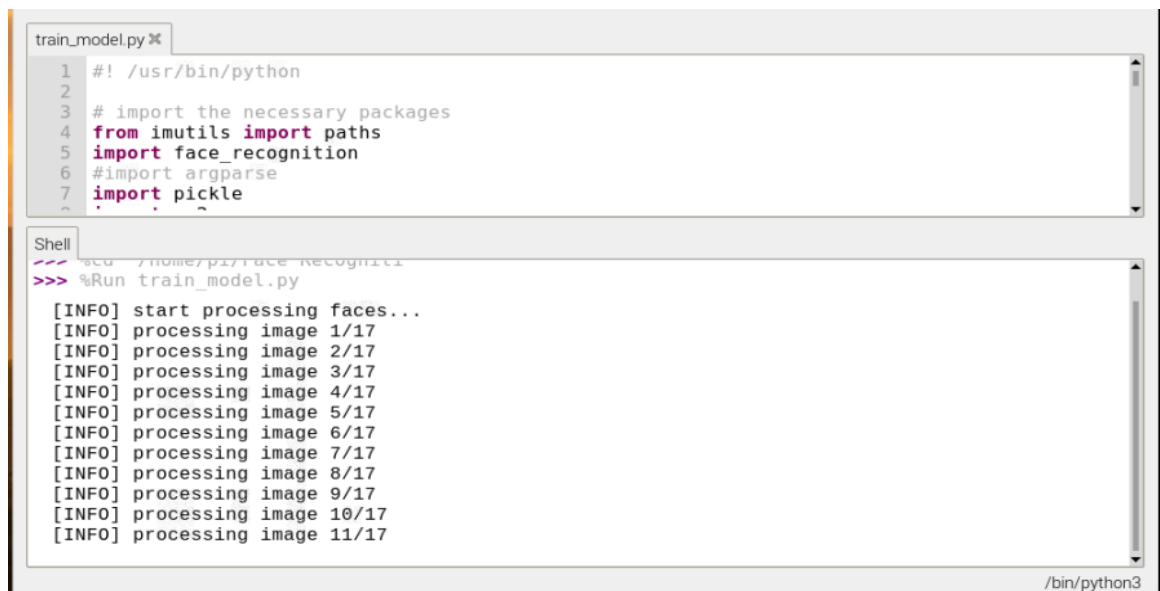
```

Thonny - /home/pi/Face Recognition/train_model.py @ 51 : 10
New Load Save Run Debug Over Into Out Stop Zoom Quit Switch to regular mode
face_shot.py train_model.py
1  #!/usr/bin/python
2
3  # import the necessary packages
4  from imutils import paths
5  import face_recognition
6  #import argparse
7  import pickle
8  import cv2
9  import os
10
11 # our images are located in the dataset folder
12 print("[INFO] start processing faces...")
13 imagePath = list(paths.list_images("dataset"))
14
15 # initialize the list of known encodings and known names
--
Shell
Python 3.7.3 (/bin/python3)
>>>
/bin/python3

```

Figure IV.15. Deuxième code

Nous exécutons ce code et notons que les données de chaque image sont enregistrées



```

train_model.py
1  #!/usr/bin/python
2
3  # import the necessary packages
4  from imutils import paths
5  import face_recognition
6  #import argparse
7  import pickle
8
Shell
~/Face Recognition
>>> %Run train_model.py
[INFO] start processing faces...
[INFO] processing image 1/17
[INFO] processing image 2/17
[INFO] processing image 3/17
[INFO] processing image 4/17
[INFO] processing image 5/17
[INFO] processing image 6/17
[INFO] processing image 7/17
[INFO] processing image 8/17
[INFO] processing image 9/17
[INFO] processing image 10/17
[INFO] processing image 11/17
/bin/python3

```

Figure IV.16. Analyse des données d'images

Le code train_model.py créer un fichier nommé encodings.pickle contenant les critères d'identification des isolateurs à étape précédente.

d) Troisième code :

Nous créons un code qui détecte l'objet en fonction de ce qui a été préalablement formé

Nous utilisons la méthode de détection HOG (Histogram of Oriented Gradients) et nous testons maintenant le modèle que nous venons d'entraîner. L'exécution du code ouvre une caméra et détecte l'objet

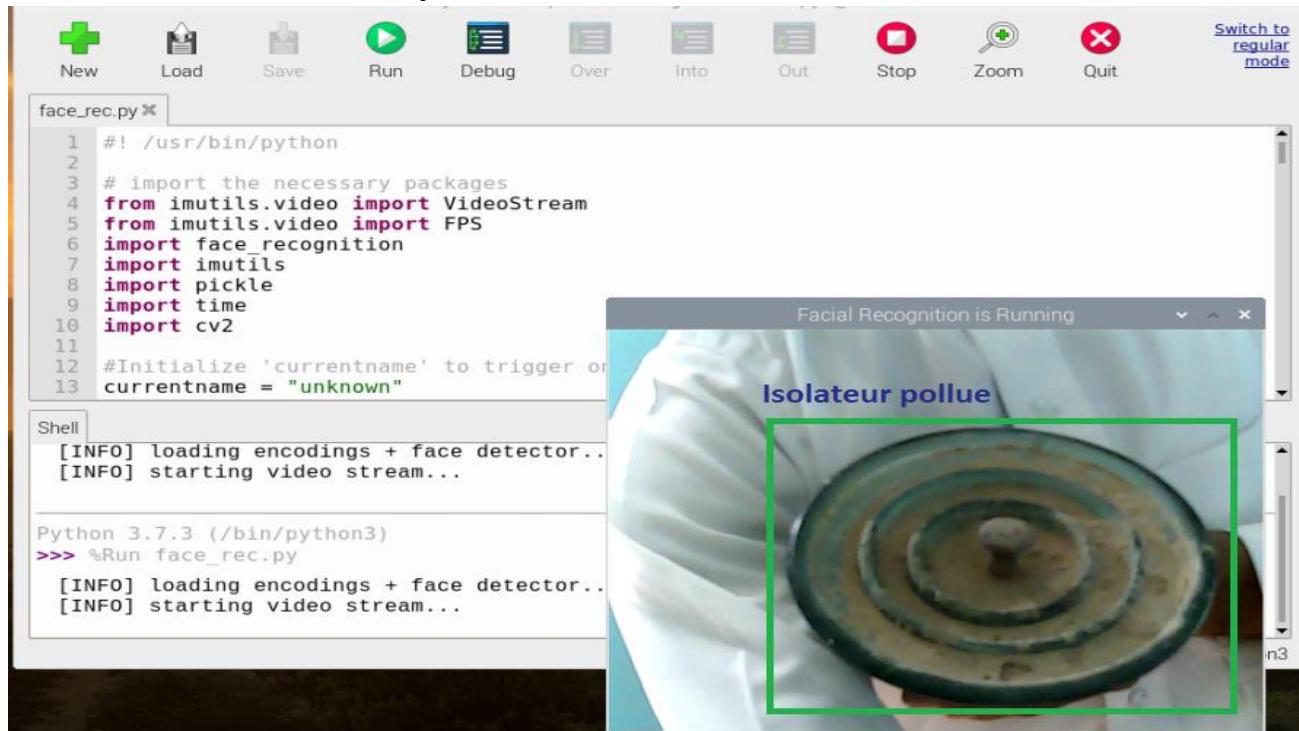


Figure IV.17. Détection d'objet par l'utilisation de Raspberry Pi.

IV.7. Conclusion :

Nous avons développé avec succès un modèle de reconnaissance de l'état de la surface d'isolateur à l'aide d'une carte à microcontrôleur Raspberry Pi et Vidéo capturée par une webcam.

Ce modèle nous permet de détecter l'état de l'isolateur qu'il soit propre ou pollué. Notre modèle Ils ont été formés grâce à des techniques d'intelligence artificielle intégrées à la bibliothèque opencv est programmé en langage python.

Les résultats obtenus montrent que notre modèle a réussi à détecter l'état des isolateurs en série et en solo.

Conclusion générale

CONCLUSION GENERALE

Au cours des dernières années, principalement en raison des progrès de l'apprentissage en profondeur, la qualité de la description d'images et de la détection d'objets a progressé à un rythme spectaculaire. La plupart de ces progrès sont le résultat d'un matériel plus puissant, des dataset (BDD) plus volumineux et de modèles plus grands et une conséquence de nouvelles idées, d'algorithmes et d'architectures réseaux améliorés. « Grâce aux Deep Learning, l'avenir de l'intelligence artificielle est prometteur ».

L'objectif principal de ce travail c'était utiliser des techniques d'intelligence artificielle pour faire le diagnostic de l'état de surface des isolateurs de haute tension à travers les outillés de détection d'objet.

La bibliothèque TensorFlow est un élément important dans la réalisation et le développement de notre projet et cela revient à la richesse de différentes fonctions qui le compose surtout dans le traitement des images et des vidéos.

Dans cette étude nous avons réalisé un model capable de détecter d'isolateurs et l'état d'isolateur est ce qu'il est propre ou pollué. Notre modèle a été entraîné à travers des techniques d'intelligence artificielle incorporé dans la bibliothèque TensorFlow programmé par le langage python ; où nous a avons compilé plusieurs images d'isolateur de haute tension en verre dans l'état propre ou pollué

Par cette méthode l'ordinateur devenu possible pour faire le diagnostic de l'état de surface des isolateurs tous simplement par les photos, les vidéos ou par un appareil photo.

Pour obtenir des résultats plus précis, il doit :

- Utilisez des plus grands nombres des photos dans la basse des données.
- Des excellentes qualités d'image utilisées
- Petite taille d'image pour un entraînement plus rapide.
- Capturez tous les côtés de l'objet à détecter.
- Adopter un maximum d'étapes pendant l'entraînement.

Dans le dernier chapitre, Raspberry Pi est devenu possible de diagnostiquer l'état de la surface des isolants simplement au caméra.

La détection d'objets est l'épine dorsale de nombreuses applications pratiques de la vision par ordinateur. Mais reste qu'on a des nombreux défis dans l'affichage des variables actuelles et toutes les difficultés auxquelles la cible est exposée. Bien que, les capacités des activités réalisées dans le domaine de la détection d'objets soient nombreuses, aucun moyen ne peut être considéré comme 100% fiable, mais sans ambages et selon la mesure, de nouveaux travaux tentent d'améliorer les scores pour obtenir de meilleurs résultats.

Bibliographer

BIBLIOGRAPHIE:

- [1] **S. D. Kimbakala**, "Modélisation dynamique des décharges se propagent sur des surfaces isolantes polluées avec des dépôts discontinus sous différentes formes de tension", Thèse doctorat, Ecole doctorale électrotechnique- électronique automatique de Lyon, France, 2007.
- [2] **N. Mhaguen**, "Développement de modèle dynamique de prédiction de la tension critique de contournement des isolateurs recouverts de glace basés sur la méthode des éléments finis ", Mémoire présenté à l'université du Québec à Chicoutimi comme exigence Partille de la maitrise en ingénierie, 2011.
- [3] **S. Tounsi, et F. Chikh**, Influence Du Sol Sur Le Comportement D'un Modèle d'Isolateur Sous tension, Mémoire Du Projet De Fin D'études Pour L'obtention Du Diplôme d'Ingénieur d'Etat En Electrotechnique, Département De Génie Electrique, Ecole Nationale Polytechnique, 2008.
- [4] **A. Bouleghlem, F. Taleb**, "Comportement de l'isolateur capot et tige 1512L artificiellement pollué sous tension alternative 50Hz", PFE, Département de Génie Electrique, ENP, Alger, Juin 2009.
- [5] **D. Dumora**, "Matériaux isolants céramiques en électrotechnique", Technique de l'ingénieur D275.
- [6] **B. Abdelghani**, Etude du développement d'arcs électriques parallèles sur une surface isolante non uniformément polluée sous tension continue, Mémoire de Magister, Université Abderrahmane Mira, Béjaia, 2009.
- [7] **J.Y. Saison**, Etude du phénomène d'humidification des dépôts naturels et artificiels de pollution sur des isolateurs électriques, Mémoire de Doctorat De Sciences, Université de Strasbourg, France, 1992.
- [8] **A. Kara**, Contribution à l'étude d'un modèle dynamique de contournement d'un isolateur pollué en électrotechnique, Université Ferhat Abbes Sétif, 2013.
- [9] **Med.A. Boudjamma, R. Sebti**, « Modélisation des phénomènes de pollution des isolateurs de haute tension », Université Kasdi Merbah-Ouargla, Master 2011-2012.
- [10] **Belhouch et Khaled**, « Estimation des paramètres de l'arc de contournement des isolateurs pollués en utilisant l'approche génétique », Mémoire de Magister, Université Ferhat Abbas Setif.
- [11] **El Bahi Azzag**, « problèmes de contournement et perforation des isolateurs de haute tension », thèse doctorat d'état, Université Badji Mokhtar, Annaba, 2007
- [12] **Hamza Slamet Amina**, « Contribution à l'étude du modèle développé d'un isolateur pollué », Mémoire de Magister en électrotechnique, université d'Oran, 2013.
- [13] **M'hamdi Ben alia**, « amélioration des performances des chaines d'isolateur de de haute tension », Thèse de Doctorat en Sciences, ENP, Alger, 2016.
- [14] **Chakou Abdelkarim Djemmaa Mohamed**, « Elaboration d'un modèle dynamique de contournement d'un isolateur pollué sous une tension impulsionnelle », mémoire de master, Université Kasdi Merbah Ouargla, 201
- [15] **F. S. Genberg**, Isolateur en céramique et en synthétique, edf-e pure, avril 1998.
- [16] **A Fatiha**, Contribution à l'étude d'un circuit équivalent des isolateurs synthétiques sous pollution en utilisant l'EMTP, Mémoire du projet, Université Ferhat Abbes Sétif, (Algerie) ;
- [17] **X. Ducouret**, lignes aériennes : matériels, techniques de l'ingénieur D4423

- [18] **Sayah Moatez Béllah** « Prédiction du contournement d'une Chaîne d'isolateur MT 175 CTV-Sonelgaz Polluée Artificiellement par la Logique Floue », mémoire de master, Université Mohamed Khider Biskra, 2014.
- [19] **M. Tegar, A. boubakeur**, « Pollution des Isolateurs de Haute Tension », ENP d'Alger, 2002.
- [20] **F. Taleb, A. Bouleghlem**, « Comportement de l'Isolateur capot et tige 15121 Artificiellement Pollué sous Tension Alternative 50 Hz » Mémoire d'Ingénieur d'Etat, ENP, Alger, 2010.
- [21] **Ahmed Seddik Bouhadjar**, Amokrane Mounir, « Caractérisation de l'état de surface d'un isolateur par technique d'ondelettes modélisation », Mémoire d'Ingénieur d'Etat, ENP d'Alger, Juin 2008.
- [22] **N Ravelomanantsoa**, Effet du vent sur le taux d'accumulation de pollution à la surface des isolateurs de haute tension dans les conditions hivernales, Mémoire de doctorat, Université du Québec à Chicoutimi, 2012.
- [23] **F. Aouabed**, Contribution à l'étude d'un circuit équivalent des isolateurs synthétiques sous pollution en utilisant l'EMTP, Mémoire magistère en électrotechnique, Université Ferhat Abbas Sétif.
- [24] **R. Bouzeroura**, Etude du développement d'arcs électriques parallèles sur une surface isolante non uniformément polluée sous tension continue, Mémoire de magister, Université de Bejaia, promotion 2009.
- [25] **N. Khairoun, E. Abderrahmane, D. Zellouta** "Essais sur isolateurs pollues et modélisation" Mémoire du projet de fin d'études laboratoire de haute tension, école nationale polytechnique d'Alger. Juin 2007/2008,
- [26] **Z. Sahli**, Etude de la Non Uniformité de la Pollution sur les Caractéristiques de Contournement des isolateurs Réels sous Tension Continue, Mémoire de Magister, Département d'électrotechnique, Université Abederrahmane Mira de Bédjaia.
- [27] **M. Ben alia**, modélisation d'un isolateur naturellement pollue sous tension impulsionnelle utilisant des circuits électriques Equivalents, mémoire de magister en génie électrique, laboratoire de haute tension, école nationale polytechnique d'Alger, 2008.
- [28] **M. Slama**, Contribution à l'étude de l'influence de la non-uniformité de la pollution basée sur la méthode de la d.d.s.e pour le dimensionnement des isolateurs des lignes t.h.t a courant alternatif, Mémoire de Magister, 29 juin 2002.
- [29]. **S. Satta**, contribution à l'étude de l'influence de la pollution discontinue sur le comportement des isolateurs : expérimentation et simulation, université Ferhat Abbas Sétif, 2011.
- [30] **P. J. Lambeth**, H. Auxel et M. P. Verma, Méthode de mesure de la sévérité de la pollution compte tenu de son influence sur le comportement des isolateurs à haute tension, Electre N°. 20, pp 95-100, Janvier 1972.
- [31] **W. Heise**, G.F.luxa, G.Revrery et M. P. Verma, Estimation de la méthode d'essais sous pollution artificielle par couche solide CIGRE, rapport 33-09 1972.
- [32] **R. Coello et B. Aladenize**, Les diélectriques, livre, Edition Hermès, Paris. 1986
- [33] **S. Moatez Béllah**, Prédiction Du Contournement D'une Chaîne D'isolateur Mt 175 Ctv-Sonelgaz Polluée Artificiellement Par La Logique Floue, Mémoire De Fin D'études En Vue De

L'obtention Du Diplôme : Master, Département De Génie Electrique, Université Mohamed Khider Biskra, 2014.

[34] **M. Malika LOUANCHI, M. Samia BELBAZ**"Exploitation du Signal du Courant de Fuite pour le Diagnostic de l'état du Réseau Electrique avec LabVIEW" Mémoire du projet de fin d'études laboratoire de haute tension, UNIVERSITE MOULOUD MAMMERI DE TIZI-OUZOU. Juin 2015/2016.

[35] <https://www.actuia.com/contribution/jeancharlesrisch/segmentation-et-detection-dobjets-entemps-reel-avec-tensorflow/>.

[36] **M.L Goff**, Techniques d'analyse de contenu appliquées à l'imagerie spatiale, Mémoire de Doctorat, Institut National Polytechnique de Toulouse, 2017

[37] **G.Bradski ET A.Kaelher**, Learning OpenCV Computer Vision with the OpenCV Library O'Reilly, Sebastopol, 2008.

[38] **H.Fouzia**, La détection et suivi des objets en mouvement dans une scène vidéo en utilisant la bibliothèque OpenCV. PhD thesis, Université Abou Bakr Belkaid– Tlemcen, 2017.

[39] <http://master-ivi.univ-lille1.fr/fichiers/Cours/pje-semaine-3-opencv.pdf>.