

UNIVERSITE DE KASDI MERBAH OUARGLA

FACULTÉ des Nouvelles Technologies de l'Information et de la Communication

DEPARTEMENT de l'Electronique et des Télécommunications



**Mémoire**

**MASTER ACADEMIQUE**

**Domaine : Science et Technologie**

**Filière : Électronique**

**Spécialité : Électronique Des Systèmes Embarqués**

**Présenté par :**

□ **BEKKARI Lakhdar**

□ **KADI Nadhir**

**Thème**

**Détection D'incendie Et De Fumée A L'aide De  
L'apprentissage Par Transfert**

**(Fire and Smoke Detection Using Transfer Learning)**

So  
ute  
nu  
pu  
bli  
qu  
em

entle : Juin 2022

Devant le jury :

□ Dr. LATI Abdelhai Encadreur/rapporteur MCBUKM OUARGLA

□ Dr. BENLAMOUDI Azeddine Président MCAUKM OUARGLA

□ Dr. CHERGUI Abdelhakim Examineur MABUKM OUARGLA

Année Universitaire : 2021 /2022

## Résumé

La dernière décennie a marqué une croissance énorme dans les domaines de l'intelligence artificielle et de la vision par ordinateur. Les progrès récents de plusieurs technologies ont permis la croissance et le développement de systèmes de surveillance intelligents grâce au développement de l'informatique de pointe et à la transmission de données sur Internet. Un système de détection d'incendie et de fumée nécessite des mécanismes de réponse précis, rapides et en temps réel pour prendre la bonne décision et avertir immédiatement le personnel correspondant. Les systèmes de détection d'incendie courants ne peuvent pas être utilisés à l'extérieur et ne garantissent pas la disponibilité nécessaire en matière de détection en temps réel. L'objectif de ce projet est d'utiliser un Deep Neural Network pour détecter le feu et la fumée dans les environnements extérieurs à l'aide d'un système embarqué avec une caméra. Récemment, après d'importants progrès dans les structures de programmation et, surtout, dans le domaine de l'apprentissage automatique, les systèmes embarqués sont désormais capables d'effectuer de l'edge computing en temps réel. Ce projet propose une approche pour effectuer la détection d'incendie et de fumée par vision par ordinateur en utilisant l'informatique de pointe. Une fois que le feu ou la fumée apparaît dans n'importe quelle caméra, le système peut détecter et envoyer immédiatement une alerte au superviseur correspondant

**Mots clés** Détection d'incendie et de fumée, Apprentissage par transfert, Réseaux légers, Edge Computing, Vision par ordinateur, Systèmes embarqués.

## ملخص

شهد العقد الماضي نموًا هائلًا في مجالات الذكاء الاصطناعي ورؤية الكمبيوتر. مكنت التطورات الحديثة في العديد من التقنيات من نمو وتطوير أنظمة المراقبة الذكية من خلال تطوير الحوسبة المتقدمة ونقل البيانات عبر الإنترنت. يتطلب نظام الكشف عن الحرائق والدخان آليات استجابة دقيقة وسريعة وفي الوقت الحقيقي لاتخاذ القرار الصحيح وإخطار الموظفين المعنيين على الفور. لا يمكن استخدام أنظمة الكشف عن الحرائق الشائعة في الهواء الطلق ولا تضمن التوافر الضروري للكشف في الوقت الفعلي. الهدف من هذا المشروع هو استخدام شبكة عصبية عميقة لاكتشاف الحريق والدخان في البيئات الخارجية باستخدام نظام مدمج به كاميرا. في الآونة الأخيرة، بعد التقدم الكبير في هياكل البرمجة، وقبل كل شيء، في مجال التعلم الآلي، أصبحت الأنظمة المضمنة الآن قادرة على أداء الحوسبة المتطورة في الوقت الفعلي. يقترح هذا المشروع نهجًا لأداء كشف الحرائق والدخان بالرؤية الحاسوبية باستخدام الحوسبة المتقدمة. بمجرد ظهور الحريق أو الدخان في أي كاميرا، يمكن للنظام اكتشاف وإرسال تنبيهه على الفور إلى المشرف المقابل

**كلمات مفتاحية** كشف الحريق والدخان، نقل التعلم، الشبكات خفيفة الوزن، الحوسبة المتطورة، الرؤية الحاسوبية،

الدمجة

الأنظمة

## Summary

The past decade has seen tremendous growth in the fields of artificial intelligence and computer vision. Recent advancements in several technologies have enabled the growth and development of intelligent surveillance systems through the development of advanced computing and data transmission over the Internet. A fire and smoke detection system requires accurate, fast and real-time response mechanisms to make the right decision and immediately notify the corresponding personnel. Common fire detection systems cannot be used outdoors and do not guarantee the necessary availability for real-time detection. The objective of this project is to use a Deep Neural Network to detect fire and smoke in outdoor environments using an embedded system with a camera. Recently, after significant progress in programming structures and, above all, in the field of machine learning, embedded systems are now able to perform real-time edge computing. This project proposes an approach to perform computer vision fire and smoke detection using advanced computing. Once the fire or smoke appears in any camera, the system can detect and immediately send an alert to the corresponding supervisor

Keywords Fire and Smoke Detection, Transfer Learning, Lightweight Networks, Edge .Computing, Computer Vision, Embedded Systems

## REMERCIEMENTS



Je tiens à exprimer ma profonde gratitude à ma famille pour

Leur amour sans bornes et mes amis pour leur soutien tout au long de ce parcours

A ma grand-mère, que Dieu ait pitié d'elle, qui a toujours attendu ces moments

Je tiens à remercier mon professeur et superviseur,

Le docteur **LATI Abdelhai**, qui m'a guidé De manière experte dans mon travail Sur  
mémoire de fin d'études notamment.

Je tiens à exprimer mes sincères remerciements au jury

Président **Mr. BENLAMOUDI Azeddine**

Examineur **Mr. CHERGUI Abdelhakim**

J'exprime ma gratitude à tout le personnel du département D'électronique  
et de télécommunications, Faculté d'électronique, À l'Université Kasdi Merbah Ouargla



# List des abréviations

- AI : Intelligence artificielle
- CNN: Convolutional Neural Networks
- CPU: central Processing unit
- DCT : Discrete cosine transforms
- DL : Deep Learning
- DNN : Deep Neural Networks
- GPU : Graphics Processing unit
- ILSVRC: ImageNet Large-Scale Visual Recognition Challenge
- IR : Infrarouge
- K-NN: k-nearest neighbors' algorithm
- ML: Machine Learning
- NAG: Nesterov accelerated gradient
- NFPA : National Fire Protection Association
- OCR : Reconnaissance optique de caractères
- OpenCV: Open Source Computer Vision Library
- RMSProp: Root Mean Square Propagation
- RVB: rouge, vert, bleu
- SGD: Stochastic gradient descen
- SIFT: Scaled Fixed Feature Transformation
- SVM : Support vector machine
- UV : Ultraviolet
- VGG: Visual geometry group
- YOLO: You Only Look Once

# Contenu

List des abréviations.....	I
Contenu.....	II
Liste des figures.....	IV
Liste des tableaux.....	V
<b>Chapiter I</b>	
I. Introduction Générale.....	1
I.1 Contexte .....	1
I.2 Motivations .....	1
I.2.1 Une Brève Chronique.....	1
I.3 Objectifs .....	3
II. Revue de littérature .....	5
II.1 Complexité du problème .....	5
II.1.1 Caractéristique du feu.....	5
II.1.2 Caractéristiques de la fumée.....	6
II.2 Une brève introduction à l'IA .....	7
II.3 Détecteurs d'incendie et de fumée traditionnels.....	10
II.3.1 Systèmes ponctuels de détection de fumée .....	10
II.3.2 Systèmes de détection de fumée par échantillonnage d'air .....	12
II.3.3 Système de détection de fumée à faisceau .....	12
II.4 Techniques de détection basées sur la Computer vision .....	13
II.4.1 Détection de couleur et de mouvement .....	13
II.5 Techniques de détection basées sur Apprentissage automatique (Machine Learning) .....	14
II.6 Techniques de détection basées sur le L'apprentissage en profondeur (Deep Learning) .....	15
<b>Chapiter II</b>	
III. Sélection et prétraitement des données .....	17
III.1 Sélection des données – Données triées sur le volet .....	18
III.2 Prétraitement des données.....	20
III.2.1 L'augmentation des données.....	20
III.3 Fractionnement des données .....	21
III.3.1 Ensemble d'entraînement.....	22
III.3.2 Jeu de validation .....	خطأ! الإشارة المرجعية غير معرفة.
III.3.3 Ensemble de test avec des scénarios réels.....	خطأ! الإشارة المرجعية غير معرفة.
III.3.4 YOLOv5 Traitement des données.....	23
<b>Chapiter III</b>	
IV. Architectures de Réseaux Neuronaux .....	25
IV.1 Réseaux de neurones profonds (Deep Neural Networks) : .....	25

IV.2	Réseau de neurones convolutif CNN .....	26
IV.2.1	Image d'entrée .....	26
IV.2.2	Convolution.....	27
IV.2.3	Couches de pooling (Pooling Layer).....	28
IV.3	Apprentissage par transfert (Transfer Learning).....	29
IV.3.1	ImageNet .....	29
IV.4	Réseaux de neurones pour l'apprentissage par transfert.....	30
IV.4.1	YOLOV5.....	30
IV.4.2	VGG16.....	32
IV.4.3	VGG 19.....	33
IV.5	Réglage des hyperparamètres.....	34
IV.6	Implémentation du réseau .....	38
Chapiter IV		
V.	Résultats expérimentaux .....	41
V.1	Indicateurs de performance .....	41
V.2	Comparaisons de performances entre les réseaux.....	43
V.2.1	Performances de VGG16 .....	44
V.2.2	Performances de VGG19 .....	46
V.2.3	Performances de YOLOv5 .....	49
V.3	Comparaison .....	51
VI.	Conclusion générale et travaux futurs.....	52
Conclusion.....		52
VII.	Références .....	53

## Liste des figures

<b>Figure 1-1:</b> Échantillon de feu et de fumée dans différents environnements .....	6
<b>Figure 1-2:</b> Hiérarchie conceptuelle de l'Intelligence Artificielle et de ses filiales .....	8
<b>Figure 1-3 :</b> Types de problèmes de vision par ordinateur .....	9
<b>Figure 1-4:</b> Schéma fonctionnel du détecteur de fumée à ionisation .....	11
<b>Figure 1-5:</b> Schéma fonctionnel du détecteur de fumée photoélectrique à diffusion de lumière. ....	11
<b>Figure 1-6:</b> Schéma fonctionnel du détecteur de fumée à prélèvement d'air .....	12
<b>Figure 1-7:</b> Schéma fonctionnel du détecteur de fumée à faisceau. ....	13
<b>Figure 2-1 :</b> Échantillons d'images de Feu .....	19
<b>Figure 2-2 :</b> Échantillons d'images de Fumée .....	19
<b>Figure 2-3:</b> Échantillons d'images Sans Feu/Sans Fumée .....	19
<b>Figure 3-1:</b> Échantillons d'images de trois classes de l'ensemble de données personnalisé.....	19
<b>Figure 3-2 :</b> Opérations aléatoires sur un exemple d'image .....	21
<b>Figure 3-3:</b> Une visualisation des divisions .....	23
<b>Figure 3-4 :</b> Représentation conceptuelle d'un réseau a Deep neural network .....	26
<b>Figure 3-5 :</b> 4x4x3 RGB Image .....	26
<b>Figure 36 :</b> Image illustre comment la convolution est effectuée sur une image d'entrée pour extraire des caractéristiques .....	27
<b>Figure 3-7 :</b> Opération de convolution sur une matrice image $M \times N \times 3$ avec un Kernel $3 \times 3 \times 3$ ...	27
<b>Figure 3-8 :</b> Illustration de la mise en commun maximale et de la mise en commun moyenne ..	28
<b>Figure 3-9:</b> Processus d'apprentissage de l'apprentissage par transfert .....	29
<b>Figure 3-10 :</b> Un échantillon de l'ensemble de données ImageNet .....	29
<b>Figure 3-11 :</b> La structure du réseau du modèle de véhicule YOLOv5 .....	32
<b>Figure 3-12 :</b> Représentation 3D de l'architecture de VGG-16 .....	32
<b>Figure 3-13 :</b> couches dans l'architecture vgg16 .....	33
<b>Figure 4-1 :</b> Représentation de l'extraction automatique de caractéristiques avec VGG-16 .....	33
<b>Figure 4-2 :</b> couches dans l'architecture vgg19 .....	34
<b>Figure 4-3 :</b> Fine tuning. ....	38
<b>Figure 4-4 :</b> Graphiques d'entraînement de VGG16.....	45
<b>Figure 4-5 :</b> Matrice de confusion de VGG16 sur mini-testset .....	46
<b>Figure 4-6 :</b> Graphiques d'entraînement de VGG19.....	47
<b>Figure 4-7 :</b> Matrice de confusion de VGG19.....	47
<b>Figure 4-8 :</b> Graphiques d'entraînement de YOLOv5 .....	49
<b>Figure 4-9 :</b> Matrice de confusion de VGG19.....	50
<b>Figure 4-10:</b> Images testées sur le réseau yolov5 .....	51

## Liste des tableaux

<b>Tableau 1</b> : Le tableau suivante montre un aperçu encore meilleur de tous les Modelés de YOLOV5 .....	31
<b>Tableau 2</b> : Matrice de confusion et formules.....	42
<b>Tableau 3</b> : Couches dans la nouvelle architecture du modèle .....	44
<b>Tableau 4</b> : Matrice de confusion de VGG16 sur mini-testset.....	45
<b>Tableau 5</b> : Couches dans la nouvelle architecture du modèle .....	46
<b>Tableau 6</b> : Matrice de confusion de VGG19 sur mini-testset.....	47
<b>Tableau 7</b> : Matrice de confusion de VGG19 sur mini-testset.....	50
<b>Tableau 8</b> : Précision, rappel et F1-Scores de différents réseaux sur mini-testset.....	50

# **Introduction Générale**

## I. Introduction Générale

### I.1 Contexte

L'intelligence artificielle est à l'origine du plus grand saut technologique que le monde n'ait pas encore connu. Comme toute autre technologie, c'est une arme à double tranchant. Mis à part les robots tueurs et les armes autonomes, les dangers de l'IA sont bien au-delà de l'imagination, elle peut manipuler les élections politiques, effectuer une discrimination algorithmique, envahir la vie privée et analyser des données confidentielles pour produire un classement social.

Ce projet présente un système de détection d'incendie et de fumée utilisant des algorithmes d'apprentissage automatique de pointe. Les États-Unis d'Amérique ont estimé que jusqu'à 10 000 vies supplémentaires pourraient être sauvées chaque année en réduisant les temps de réponse du 911 d'une minute seulement. L'objectif de ce projet est de contribuer à la route secteur de la sécurité en effectuant une détection en temps réel des incendies et des fumées.

### I.2 Motivations

#### I.2.1 Une Brève Chronique

Le feu a joué un rôle important dans le progrès de la civilisation humaine. Au contraire, c'est l'une des catastrophes majeures entraînant d'énormes pertes en vies humaines et en biens sur toute la planète. La détection précoce des incendies peut aider à alerter et prévenir ces catastrophes, sauver à la fois des vies et des biens. Habituellement, la combustion des objets commence par de la fumée avant de prendre feu. Ainsi, la présence de fumée peut être utilisée comme un indicateur pour détecter un incendie plus tôt. Dans les environnements intérieurs, la fumée et Les détecteurs de flamme sont largement utilisés pour les alarmes incendie. Cependant, ces capteurs physiques traditionnels ont leurs limites. Ces capteurs ne fonctionnent qu'à proximité et ne seront pas assez efficaces dans les grands espaces ou les environnements extérieurs.

Selon la National Fire Protection Association (NFPA), les services d'incendie des États-Unis ont répondu à environ 1 318 500 incendies en 2018. Ces incendies ont fait 3 655 morts parmi les civils ; 15 200 civils blessés par le feu ; et environ 25,6 milliards de dollars de pertes directes sur les biens (ce chiffre comprend 12,4 milliards de dollars de pertes dues aux principaux incendies de forêt en Californie). En moyenne, il y a eu un civil tué par le feu toutes les 2 heures et 24 minutes et un civil blessé par le feu toutes les 35 minutes en 2018. Les incendies domestiques ont causé 2 720, soit 74 %, des civils tués par le feu. Les incendies représentaient 4 % des 36 746 500 appels totaux. Huit pour cent des appels étaient de fausses

## Introduction Générale

---

alarmes ; 64% des appels concernaient une aide médicale telle que des services médicaux d'urgence (EMS) ou des secours [1]

Les feux de forêt en Algérie 2021 sont des feux de forêt qui se sont déclenchés depuis le début du mois de juillet 2021, alors que l'est de l'Algérie a connu de grands incendies qui ont dévoré des dizaines d'hectares, notamment dans la région des Aurès, et de multiples feux de forêt en Algérie jusqu'au soir du 11 août 2021 ont fait 69 morts personnes, dont 28 militaires et 41 civils Au moins 12 autres personnes ont été blessées, en plus de l'incendie de dizaines de maisons et de l'incendie de forêts d'oliviers, source de subsistance des habitants de Tizi Ouzou. Une deuxième vague a éclaté le lundi 9 août 2021 avec un total de 71 incendies (soit plus de 100) brûlant dans 18 États du pays.

Pour réduire ces catastrophes, il est crucial de détecter un incendie à un stade précoce sans aucune fausse alarme. Comme les systèmes de détection traditionnels se sont avérés inefficaces dans les environnements extérieurs, diverses technologies de détection d'incendie autonomes sont en cours de développement et sont largement utilisées dans la vie réelle.

Avec l'essor récent de l'intelligence artificielle et de ses filiales, plusieurs systèmes de détection basés sur la vision par ordinateur ont vu le jour. Ces systèmes de détection basés sur la vision surmontent les limitations de taille, d'emplacement et ont une plus grande couverture de surveillance. Ils offrent également une détection plus rapide, moins de fausses alarmes et réduisent l'intervention humaine. Les chercheurs ont investi des efforts considérables pour résoudre les problèmes de complexité du système et de fausse détection en termes de technologie de vision par ordinateur. Les utilisations de la technologie pour de mauvaises intentions sont innombrables. Bien avant que le vrai travail sur l'IA ne commence,

Les gens utilisaient des inventions et des découvertes pour leurs applications involontaires. Par exemple, le type de production d'énergie connu le plus puissant est l'énergie nucléaire. Bien canalisé, il peut éclairer les villes et faciliter la vie de millions d'humains. De telles quantités d'énergie peuvent également être conçues pour être dissimulées dans un conteneur et libérées en une fraction de seconde, ce qui en fait une bombe nucléaire qui réduit les villes en poussière.

Pour en revenir au monde moderne les gens ont déjà commencé à ressentir les futurs impacts négatifs que l'IA pourrait apporter. La Chine a déjà commencé à utiliser des logiciels d'IA et des outils d'apprentissage automatique qui étudient le comportement des consommateurs pour prédire leurs goûts et leurs préférences. Ces informations peuvent ensuite être utilisées pour

le classement social qui détermine si un certain citoyen a le droit de demander des hypothèques ou des visas ou des prestations sociales. De telles histoires et leçons motivent l'utilisation de la technologie, en particulier de l'IA, de manière pacifique et bénéfique. Qu'elle soit humaine ou artificielle, l'Intelligence visant les meilleures intentions se traduira par un monde meilleur. En utilisant l'IA pour détecter le feu et la fumée, ce projet vise à rendre le monde meilleur.

### **I.3 Objectifs**

De manière générale, ce projet vise à utiliser des algorithmes d'apprentissage automatique de pointe pour servir les communautés. Le temps de réponse des unités d'urgence est probablement le facteur numéro un qui détermine si quelqu'un s'en sortira vivant ou non.

Ce projet utilise un système de surveillance autonome en temps réel qui détecte le feu et la fumée à l'aide de l'intelligence artificielle et de l'apprentissage automatique. Le but ultime de ce projet serait de réduire le taux de mortalité routière en détectant les anomalies, principalement les incendies et les fumées, et en les signalant aux autorités correspondantes.

# Chapitre I

Revue de littérature

## II. Revue de littérature

### Introduction

Les systèmes de détection d'incendie et de fumée jouent un rôle essentiel dans la détection rapide de l'incendie, avec précision et en fournissant des notifications d'alerte précoce. Il est également important d'envoyer une notification rapide dès que possible pour prévenir les dommages et économiser au autant de vies que possible. Plusieurs systèmes de détection allant de la détection traditionnelle des systèmes qui utilisent des composants mécaniques aux derniers appareils qui tirent parti des progrès technologiques récents tels que la vision par ordinateur, l'apprentissage automatique, et l'apprentissage en profondeur sont passés en revue.

Ce chapitre donne un aperçu des problèmes existants liés au feu et à la fumée, et la recherche pertinente disponible pour trouver la solution pour sa détection ainsi que les derniers paradigmes.

### II.1 Complexité du problème

Le feu et la fumée sont principalement des sous-produits l'un de l'autre. Les fumées de fumée peuvent ou non se transformer ou allumer leur source, ce qui entraîne des flammes, mais la flamme génère toujours des fumées. Il est difficile de dire si une fumée peut produire du feu, mais le feu produira toujours de la fumée avec lui. Bien que ces deux éléments soient inséparables, leurs caractéristiques sont toujours séparables l'une de l'autre. Ces caractéristiques aideront à comprendre le feu et la fumée d'une manière appropriée pour dessiner et concevoir des caractéristiques pour les détecter.

#### II.1.1 Caractéristique du feu

La forme de toute flamme est à peu près triangulaire, large à la base et se rétrécissant vers la pointe. Ceci est influencé par des facteurs tels que la vitesse et la direction du vent et la disponibilité des agents de combustion. La géométrie de la flamme évolue en fonction de la géométrie et de la surface de la source (origine de la combustion). La taille, la surface, la forme et le nombre de zones de feu dans une image varient d'une image à l'autre.

Forme La forme de toute flamme est à peu près triangulaire, large à la base et se rétrécissant vers la pointe. Ceci est influencé par des facteurs tels que la vitesse et la direction du vent et la disponibilité des agents de combustion. La géométrie de la flamme change en fonction de la géométrie et de la surface de la source (origine de la combustion). La taille, la surface, la forme et le nombre de zones d'incendie dans une image varient d'une image à l'autre

Couleur Le feu possède des couleurs et des lueurs complexes, et produit souvent de la suie qui est une substance poudreuse noire produite par la combustion incomplète de matière organique. Les particules de suie émises par la source de combustion se comportent comme des corps noirs parfaits. Ces particules de suie produisent une lueur rouge-orange du feu. La combustion complète du gaz a une couleur bleu pâle en raison de l'émission d'un rayonnement à une seule longueur d'onde. Une grande partie du rayonnement est émise dans les bandes visible et infrarouge. La couleur dépend de la température pour le rayonnement du corps noir et de la composition chimique pour les spectres d'émission. La couleur dominante d'une flamme change avec la température. La base du feu sera de couleur blanche, la couleur la plus chaude possible pour les matières organiques, puis passera progressivement à la région de couleur jaune, puis aux régions orange et rouge. Ces couleurs peuvent ou non correspondre aux températures de chaleur rouge/orange/jaune/blanc selon la composition chimique de leur source. Par exemple, toute source de feu contenant du nitrate de baryum brûle un vert vif qui n'est pas une couleur régulière dans la résolution sauvage.



Figure1- 1: Échantillon de feu et de fumée dans différents environnements

### II.1.2 Caractéristiques de la fumée

La fumée est composée de particules en suspension dans l'air et de gaz émis lorsqu'un matériau subit une combustion ou une pyrolyse, ainsi que de la quantité d'air qui est entraînée ou autrement mélangée à la masse. La composition de la fumée dépend de la nature du combustible en combustion et des conditions de combustion. Plus la quantité d'oxygène disponible est élevée, plus la quantité de fumée produite est faible lorsque la source brûlante a tendance à brûler complètement avec le feu

**Forme** La forme de la fumée est plus concentrée à la base de son origine et se diffuse à mesure qu'elle monte dans le milieu environnant. Le panache de fumée monte généralement droit ou s'incline vers le haut sous l'effet de la flottabilité et du vent. Son diamètre augmente

constamment en formant une région conique inversée. L'orientation mobile de la fumée varie continuellement, ce qui entraîne à son tour une texture variable continue.

**Couleur** Les pompiers examinent généralement la fumée pour comprendre le scénario actuelle d'un incendie. Une caractéristique importante pour prédire LE comportement du feu est la couleur la fumée émise par la source. La couleur de la fumée indique le quand et la densité des combustibles impliqués, qui donnent tous des indices sur ce que le feu pourrait faire ensuite. La fumée blanche signifie souvent que le matériau dégage de l'humidité et de la vapeur d'eau, ce qui indique que la feu star tout juste à consommer le matériau. La fumée blanche peut également indiquer des combustibles légers et cendrés tels que l'herbe ou Les brindilles. Une fumée épaisse et noire indique des carburants lourds qui ne sont pas entièrement consommés. Parfois, la fumée noire peut être UN indicateur qu'un matériau artificiel brûle comme des pneus, des véhicules ou une structure. L'invar unit règle générale, and la fumée est foncée, plus le feu est volatil. Une fumée grise peut indiquer que le feu ralentit et qu'il ambitions de matière à brûler.

## II.2 Une brève introduction à l'IA

**L'intelligence artificielle** vise à sonder et à créer des entités intelligentes. Avec le développement rapide de la technologie et la disponibilité étendue des données numériques dans toutes les industries, une immense révolution dans les tendances des données a entraîné le développement de nouveaux algorithmes et technologies pour traiter et résoudre de nombreux problèmes de l'ère moderne. L'apprentissage automatique est l'une des branches de l'intelligence artificielle (IA) qui permet aux machines d'apprendre de l'environnement et de leur expérience contrairement à la programmation explicite pour des cas d'utilisation spécifiques.

**Apprentissage automatique (Machine Learning)**L'apprentissage automatique est une application de l'intelligence artificielle qui fournit aux systèmes la capacité d'apprendre et de s'améliorer automatiquement à partir de l'expérience sans être explicitement programmés. Il se concentre principalement sur le développement de programmes et de modèles sophistiqués difficiles à réaliser par les méthodes traditionnelles. Il facilite également l'analyse d'énormes quantités de données. Cette technique fournit également des résultats beaucoup plus rapides et efficaces pour signaler des opportunités lucratives ou des risques dangereux sur certains problèmes. Combiner l'apprentissage automatique avec l'IA et les technologies cognitives peut le rendre encore plus efficace dans le traitement de gros volumes d'informations[2]

**L'apprentissage en profondeur (Deep Learning)**est une approche d'apprentissage automatique pour apprendre différents niveaux de représentations ainsi que des fonctions de cartographie et intègre ces concepts de manière hiérarchique. Ceci peut être réalisé en

décomposant des modèles informatiques ou des graphiques en plusieurs couches de traitement, dans le but d'apprendre des représentations des données avec plusieurs niveaux d'abstraction. Les modèles d'apprentissage en profondeur peuvent apprendre des fonctionnalités de bas niveau à partir de données brutes et peuvent apprendre des et différencier les caractéristiques avec leur complexité accrue des données traitées. L'apprentissage en profondeur a été appliqué à des domaines tels que la vision par ordinateur, la reconnaissance vocale, le traitement du langage naturel, la reconnaissance audio, le filtrage des réseaux sociaux, la traduction automatique, la bio-informatique, la conception de médicaments, l'analyse d'images médicales, l'inspection de matériaux et les programmes de jeux de société, où ils ont produit des résultats comparables. À et dans certains cas supérieur aux experts humains[3]

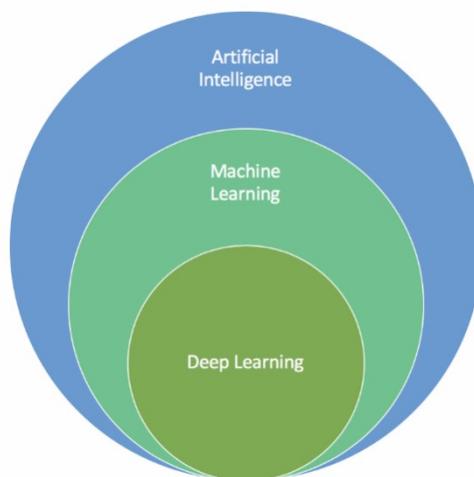
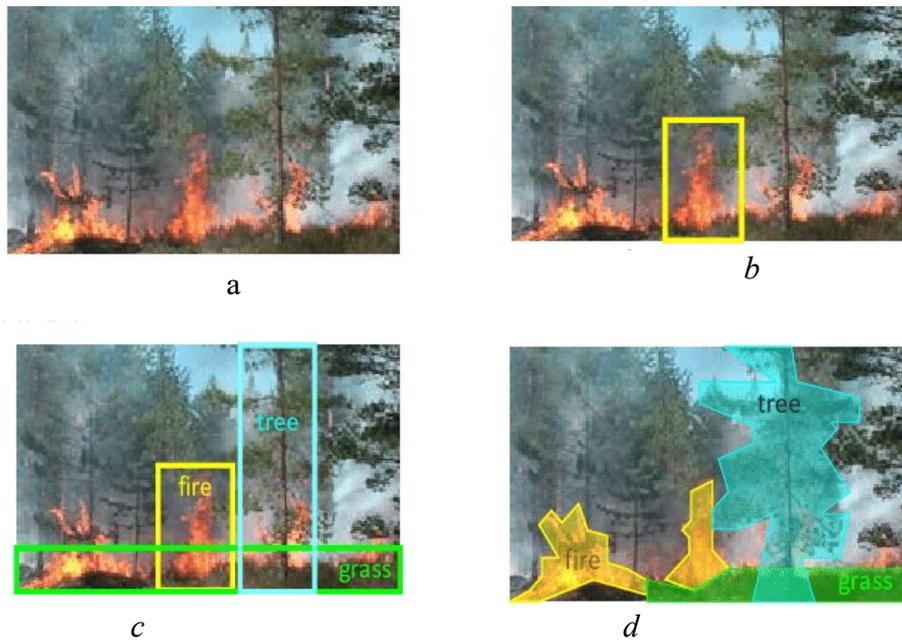


Figure 1-2: Hiérarchie conceptuelle de l'Intelligence Artificielle et de ses filiales [4]

**La vision par ordinateur** est l'un des outils les plus importants de l'IA. La vision par ordinateur est un domaine scientifique interdisciplinaire de l'informatique qui se concentre sur la façon dont les ordinateurs peuvent être conçus pour acquérir les capacités de la vision humaine dans une compréhension de haut niveau du contenu des images numériques ou des vidéos[5]. On pense que ce domaine ne peut que travailler avec une capacité limitée en raison de l'indisponibilité de vastes calculs et données. Les progrès récents de l'IA, de l'apprentissage automatique et des réseaux de neurones, ainsi que la quantité de données disponibles, se sont avérés efficaces avec la vision par ordinateur pour faire un grand pas en avant en surpassant les humains dans certaines tâches liées à la détection et à l'étiquetage d'objets. Ce domaine a connu un énorme succès dans la résolution des problèmes liés à la reconnaissance optique de caractères (OCR), à l'inspection des machines, à la construction de modèles 3D, à l'imagerie médicale, à la sécurité automobile, à la capture de mouvement (mo cap), à la surveillance, à la reconnaissance des empreintes digitales et à la biométrie.



*Figure 1-3 : Types de problèmes de vision par ordinateur [6].*

La vision par ordinateur est utilisée de plusieurs façons pour résoudre différents problèmes. Certains de ces problèmes sont représentés dans la figure suivante.

La figure 5 représente quatre principaux types de problèmes de vision par ordinateur :

- A. Classement
- B. Classification + Localisation
- C. Détection multi-objets
- D. Segmentation sémantique

### II.3 Détecteurs d'incendie et de fumée traditionnels

Cette section traite de certains des systèmes traditionnels qui utilisent des capteurs mécaniques et électroniques pour détecter un incendie ou de la fumée dans l'environnement environnant. La plupart de ces systèmes de capteurs sont destinés à un usage intérieur.

#### II.3.1 Systèmes ponctuels de détection de fumée

Ces systèmes utilisent des dispositifs qui réagissent aux particules de fumée produites par un incendie. Ils fonctionnent selon le principe de fonctionnement par ionisation, photoélectrique, chambre à brouillard ou autre analyse des particules de fumée. Les détecteurs de fumée de type ponctuel utilisent soit le principe de fonctionnement à ionisation, soit le principe photoélectrique. Ces systèmes sont destinés à l'alerte précoce. Certains sont conçus pour être installés dans des conduits de ventilation. Correctement installés, les détecteurs de fumée peuvent détecter

Les particules de fumée aux tout premiers stades d'un incendie dans les zones où ils se trouvent[7]. Ces systèmes de détection sont classés en deux sous-systèmes en fonction de leurs différents principes de fonctionnement et sont largement utilisés dans les systèmes de détection d'intérieur.

Détection de fumée par ionisationLe principe de l'utilisation d'une petite quantité de matière radioactive pour ioniser l'air entre deux électrodes chargées différemment pour détecter la présence de particules de fumée. Les particules de fumée entrant dans le volume d'ionisation diminuent la conductance de l'air en réduisant la mobilité des ions. Le signal de conductance réduite est traité et utilisé pour transmettre une condition d'alarme lorsqu'il répond à des critères prédéfinis. La détection de fumée par ionisation est plus sensible aux particules invisibles (taille inférieure à 1 micron) produites par la plupart des feux enflammés. Il est un peu moins sensible aux particules plus grosses typiques de la plupart des feux couvant[8]

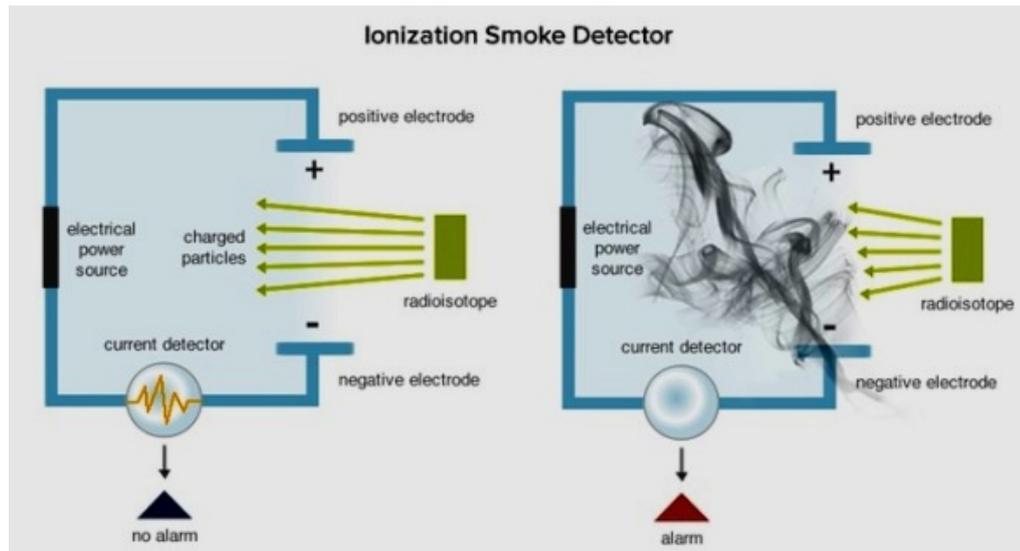


Figure 1-4: Schéma fonctionnel du détecteur de fumée à ionisation [7]

**Détection de fumée par diffusion de lumière photoélectrique** Ce système de capteur relève de la classe de détection optique, utilise le principe d'une source lumineuse et d'un capteur photosensible agencé de sorte que les rayons de la source lumineuse ne tombent normalement pas sur le capteur photosensible. Lorsque des particules de fumée obstruent le trajet de la lumière, une partie de la lumière est diffusée par réflexion et réfraction sur le capteur. Le signal lumineux est traité et utilisé pour transmettre une condition d'alarme lorsqu'il répond à des critères prédéfinis. La détection photoélectrique de fumée par diffusion de lumière est plus sensible aux particules visibles (d'une taille supérieure à 1 micron) produites par la plupart des feux couvant. Il est un peu moins sensible aux particules plus petites typiques de la plupart des feux enflammés. Il est également moins sensible à la fumée noire qu'à la fumée de couleur plus claire [10]

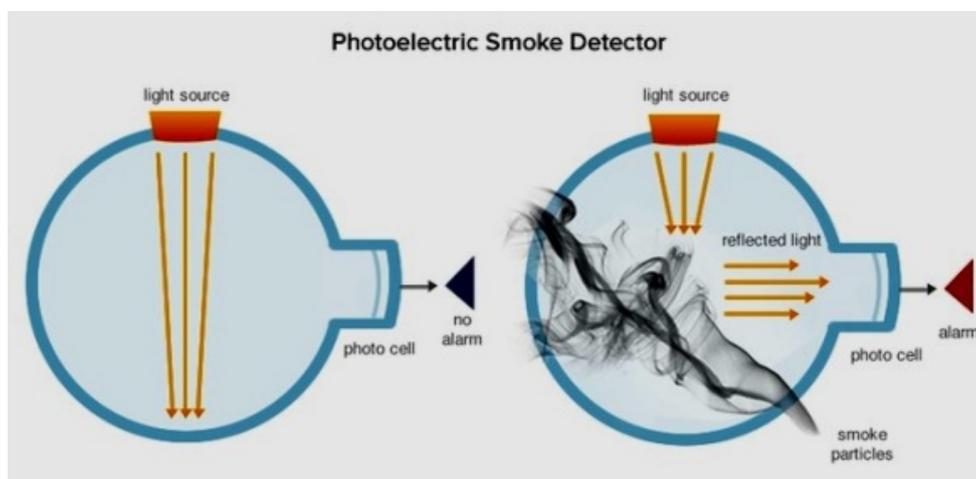


Figure 1-5: Schéma fonctionnel du détecteur de fumée photoélectrique à diffusion de lumière [8]

### II.3.2 Systèmes de détection de fumée par échantillonnage d'air

Un détecteur d'échantillonnage d'air, également connu sous le nom de détecteurs de fumée par aspiration, consiste en un réseau de distribution de tuyaux ou de tubes qui va du détecteur à la ou aux zones à protéger. Un ventilateur d'aspiration dans le boîtier du détecteur aspire l'air de la zone protégée vers le détecteur via des orifices d'échantillonnage d'air, des canalisations ou des tubes. Au niveau du détecteur, l'air est analysé pour détecter les produits du feu. Les détecteurs de fumée à prélèvement d'air sont capables de détecter un incendie à son stade le plus précoce. Les détecteurs traditionnels « de type ponctuel » détectent passivement la fumée ou la chaleur dans leur environnement immédiat. Les détecteurs de fumée à prélèvement d'air aspirent activement l'air ambiant à travers un réseau de canalisations jusqu'à sa chambre de détection où il peut détecter la présence de particules créées aux tout premiers stades de la combustion, avant même que la fumée ne soit visible. En fonction de la façon dont ils utilisent l'échantillon d'air, ceux-ci sont divisés en trois types[12].

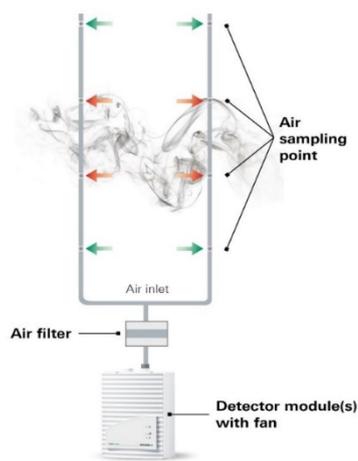
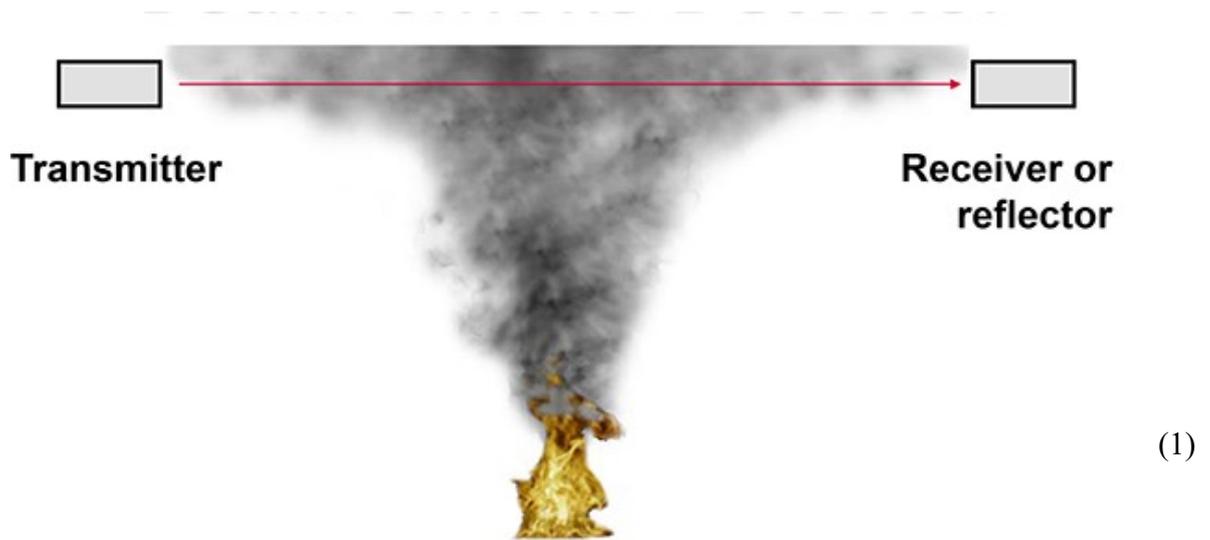


Figure 1-6: Schéma fonctionnel du détecteur de fumée à prélèvement d'air [9]

### II.3.3 Système de détection de fumée à faisceau

La détection de faisceau est devenue courante dans les grands espaces ouverts, tels que les atriums, les entrepôts, etc. Les détecteurs de faisceau sont rentables par rapport à l'utilisation de la détection ponctuelle et conviennent à des hauteurs bien au-dessus des limites de fonctionnement de la détection de fumée conventionnelle. Cet appareil détecte la fumée en projetant un faisceau lumineux d'un émetteur-récepteur à travers la zone protégée vers un réflecteur qui renvoie le signal lumineux vers l'émetteur-récepteur. La fumée entrant dans le trajet du faisceau diminuera le signal lumineux provoquant une alarme[9].



**Figure 1-7:** Schéma fonctionnel du détecteur de fumée à faisceau [1].

Certains systèmes détectent la fumée et le feu à l'aide de différents mécanismes tels que la détection de conduit, la détection de chaleur localisée, la détection de chaleur linéaire, la détection de gaz et la détection de flamme à l'aide de longueurs d'onde infrarouge (IR), ultraviolet (UV). Ces systèmes de détection ne sont pas pertinents à ce sujet car ils utilisent la conduction thermique comme principe de fonctionnement, détectent l'augmentation lente et rapide de la température dans l'environnement environnant. Cependant, il existe un autre système de détection de flamme à l'aide de caméras numériques qui sera abordé dans la section suivante.

## II.4 Techniques de détection basées sur la Computer vision

Des algorithmes de vision par ordinateur pour la détection vidéo automatique d'incendie ou de fumée ont été développés pour des applications dans les tunnels, les hangars d'avions, les navires, etc. Ils sont principalement dédiés aux petits espaces encombrés. Une quantité importante de recherches s'est concentrée sur le développement de systèmes de détection d'incendie vidéo fiables dans un environnement vaste ou ouvert. Plusieurs systèmes de détection d'incendie basés sur la vision par ordinateur fonctionnent en extrayant la couleur et le mouvement qui sont deux caractéristiques importantes

### II.4.1 Détection de couleur et de mouvement

Modèle de couleur RVB Ce modèle fonctionne en extrayant des informations des pixels de feu dans l'image en déterminant la composante de couleur du pixel. Trois composants déterminent la couleur d'un pixel : R (rouge), V (vert) et B (bleu). D'après [70], on considère qu'un pixel est un pixel feu si les valeurs de chaque composante respectent les relations :

$$R > G > B$$

Pour la détection d'incendie, la composante R est considérée comme la composante dominante ; en ce sens, on impose que R ait un seuil déterminé. Ainsi, un masque peut être créé et il peut être utilisé pour marquer les pixels de feu dans une image. Cette méthode fonctionne sur les formules suivantes pour créer le masque.

$$R1(x, y) = (R(x, y) > G(x, y)) \&\& (G(x, y) > B(x, y))$$

$$R2(x, y) = (R(x, y) > 190) \&\& (G(x, y) > 100) \&\& (B(x, y) < 140)$$

$$M = R1 \&\& R2$$

## II.5 Techniques de détection basées sur Apprentissage automatique (Machine Learning)

Des algorithmes d'apprentissage automatique sont utilisés pour trouver une relation entre les données non conventionnelles et leurs composants à l'aide de fonctions non linéaires. Ces algorithmes sont formés sur des données pour apprendre les caractéristiques (2) utilisant des mécanismes d'apprentissage supervisés ou non supervisés. Quelques-unes (3) techniques liées à la détection d'incendie et de fumée ont été étudiées et répertoriées ici. (4)

Jing et al. (2008)[15], ont proposé un algorithme de détection de mouvement simple, après quoi la propriété d'irrégularité de la fumée est utilisée comme ensemble de caractéristiques. Certains autres paramètres tels que la zone de premier plan, le nombre de blocs de fumée, le périmètre de la fumée et la zone de la fumée sont également introduits afin de générer des vecteurs de caractéristiques. La machine à vecteurs de support (SVM) est ensuite finalement utilisée pour la classification. Cette méthode est signalée pour obtenir des performances satisfaisantes, mais le problème du taux de fausses alarmes doit encore être résolu.

Gubi et al. (2009) [11] ont proposé une approche de détection de fumée extérieure basée sur les ondelettes et la machine à vecteurs de support (SVM). Dans le schéma, les cadres d'image sont divisés en petits blocs de 32 x 32 pixels et la caractérisation de la fumée est effectuée à l'aide d'une

approche standard de reconnaissance de formes avec des sous-unités de prétraitement, d'extraction de caractéristiques et de classification avec des phases d'apprentissage et de test. Dans l'étude, les transformées discrètes en cosinus (DCT) et les transformées en ondelettes sont utilisées pour générer des caractéristiques ; et les caractéristiques extraites sont testées séparément à l'aide d'un classificateur linéaire K-NN et d'un classificateur non linéaire SVM pour étudier leurs performances respectives. On constate que la combinaison des ondelettes et du

classificateur de vecteur de support donne la plus faible incidence de faux positifs, la sensibilité la plus élevée de 0,9 et la spécificité de 0,89. Un total de 60 caractéristiques pour trois niveaux est obtenu et utilisé comme entrée dans un classificateur binaire SVM basé sur une fonction radiale. Bien que des résultats très prometteurs soient obtenus, la grande dimension du vecteur caractéristique pourrait rendre la méthode trop lente pour un schéma de détection pratique.

## **II.6 Techniques de détection basées sur le L'apprentissage en profondeur(Deep Learning)**

Plusieurs domaines de l'apprentissage automatique ont connu des progrès significatifs avec l'introduction de l'apprentissage en profondeur. Les algorithmes d'apprentissage en profondeur ont été appliqués avec succès à divers domaines tels que la détection/classification d'objets dans les images, la reconnaissance vocale et le traitement du langage naturel. Les chercheurs ont mené diverses études sur la détection d'incendie et de fumée basées sur l'apprentissage en profondeur pour améliorer les performances et l'efficacité. Certaines de ces recherches ont été répertoriées et étudiées pour comprendre le problème

Dongqing Shen et al. (2018) [17], ont proposé d'utiliser le modèle YOLO pour mettre en œuvre la détection de flamme. You Only Look Once (YOLO) est un modèle en temps réel pour la détection d'objets. Cette méthode utilisait un réseau YOLO simplifié avec neuf couches de convolution et une couche supplémentaire pour la préformation. Chaque couche de convolution sera suivie de la couche max-pooling et d'une fonction d'activation ReLU. L'incendie dans l'image a été détecté à l'aide des boîtes englobant. Le modèle de détection de flamme proposé a obtenu une précision de 76% sur leur ensemble de test.

# Chapitre II

**Sélection et prétraitement des données**

### III. Sélection et prétraitement des données

Les données jouent un rôle essentiel dans les performances de toute architecture de réseau et la sélection des fonctionnalités à partir des données existantes est importante tout en résolvant tout problème d'apprentissage à portée de main. La sélection des fonctionnalités, le processus de recherche et de sélection des fonctionnalités les plus utiles dans un ensemble de données, est une étape cruciale du pipeline d'apprentissage automatique. Tout algorithme ou topologie de réseau doit traiter la bonne quantité de données pertinentes pour obtenir les bonnes fonctionnalités. Ces caractéristiques appartenant aux données

Représentent le problème et nous aident à le résoudre en prédisant ces modèles de caractéristiques similaires sur des données invisibles. Un ensemble spécial de données étiquetées fournissant des informations connues est utilisé dans l'apprentissage supervisé pour construire un modèle de classification ou de régression. Un algorithme d'apprentissage supervisé déduit une fonction de classification ou de régression à partir de l'ensemble de données donné. La fonction déduite doit alors prédire une valeur de sortie appropriée pour un vecteur de caractéristiques d'entrée donné. L'objectif de la phase d'apprentissage est d'estimer les paramètres d'un modèle pour prédire les valeurs de sortie avec une bonne performance dans des applications réelles de ce modèle d'apprentissage automatique.

Au cours du processus d'apprentissage, la plupart des algorithmes utilisent toutes les instances de l'ensemble de données donné pour estimer les paramètres du modèle, mais généralement, de nombreuses instances de l'ensemble d'apprentissage sont inutiles. Ces instances ne peuvent pas améliorer les performances prédictives du modèle et peuvent même les dégrader et sont considérées comme une charge pour tout réseau. Il y a plusieurs fonctionnalités à ignorer qui font dévier l'algorithme de la bonne courbe d'apprentissage. Il est donc important de savoir que les instances et les fonctionnalités aideront le modèle à mieux apprendre et à bien fonctionner dans des scénarios réels.

La taille et la structure de l'ensemble de données doivent estimer correctement les paramètres d'un modèle et peuvent varier d'un problème à l'autre et d'une méthode de sélection de caractéristiques choisie. Un grand ensemble de données peut sur-ajuster le modèle tandis que de petits ensembles de données peuvent sous-ajuster le modèle, ce qui rend le modèle inutile lors de l'exécution sur des données plus générales dans des applications réelles. Il est également important de comprendre un autre fait, à savoir qu'un grand ensemble de données avec un

nombre de fonctionnalités non pertinent et moindre, un ensemble de données plus petit avec des fonctionnalités riches modifiera également les performances du modèle.

Avec la bonne taille et la bonne structure, des ensembles de données enrichis et généralisés doivent être choisis pour apporter les meilleures performances avec le réseau et l'algorithme donnés du modèle d'apprentissage automatique.

Le chapitre 3 traite des ensembles de données sélectionnés, des opérations de prétraitement appliquées pour enrichir l'ensemble de données avec les bonnes fonctionnalités et de l'ensemble de données final pour former et valider le modèle pour une application réelle sur l'environnement extérieur.

### **III.1 Sélection des données – Données triées sur le volet**

Même après des recherches approfondies et diverses implémentations sur la détection des incendies et de la fumée, peu de données sont librement disponibles. La plupart des implémentations existantes ont fonctionné sur des ensembles de données de vidéos et toutes ces données ne peuvent pas être utilisées car les images obtenues grâce à cette approche sont une série d'images de ces vidéos. Le réseau doit apprendre les caractéristiques de l'environnement extérieur ainsi que les événements d'incendie et de fumée à partir de la vue de la caméra de surveillance.

Le réseau doit également comprendre l'environnement extérieur où il n'y a pas d'événement d'incendie ou de fumée dans la scène. Il peut y avoir divers événements naturels dans les environnements extérieurs qui peuvent ressembler à du feu ou de la fumée. Des événements tels que le lever et le coucher du soleil, les reflets sur les surfaces par temps clair et les phares des véhicules la nuit. De tels événements peuvent ressembler à un incendie alors qu'un environnement couvert, neigeux et brumeux peut ressembler à de la fumée. Tous ces événements doivent être traités correctement car ces cas peuvent déclencher de fausses alarmes dans les applications en temps réel. L'ensemble de données doit traiter toutes ces images pour que le réseau connaisse leurs caractéristiques et détecte ces cas comme des événements sans incendie ou sans fumée.

Le réseau est censé détecter les incendies et la fumée dans les zones industrielles, les zones routières et les calamités naturelles. L'ensemble de données doit contenir suffisamment de détails sur ces événements et représenter une bonne quantité d'éléments de feu et de fumée dans la scène pour être visibles sur la caméra. Il est en outre important de connaître la diversité de ces caractéristiques telles qu'une couleur, une forme et un arrière-plan différents pour une

classification appropriée des incidents d'incendie et de fumée, en particulier lorsqu'il s'agit d'incendies dans des applications en temps réel pouvant être accompagnés de fumées denses qui sont plus visibles que les flammes de la source d'incendie ou parfois seules des fumées fines et de couleur pâle sont celles visibles de la source., l'attribut le plus dominant doit être pris en compte pour l'apprentissage des fonctionnalités.

Il est nécessaire de comprendre le pourcentage d'occupation de ces éléments dans les images du jeu de données. La sélection manuelle des données tout en gardant tous ces points sous contrôle avant de procéder à la formation du réseau pourrait aider les couches du réseau à extraire et interpréter les bonnes informations à partir des caractéristiques, ce qui se traduit par un modèle bien appris. Après une enquête approfondie sur plusieurs ensembles de données, deux semblaient très impressionnants.

Cet ensemble de données personnalisé contient des images représentant les trois catégories de feu, de fumée et de non-feu et non-fumée. Les images sont soigneusement sélectionnées pour représenter les bonnes informations sur les caractéristiques à apprendre par le réseau.



A. Échantillons d'images de Feu



B. Échantillons d'images de Fumée



C. Échantillons d'images Sans Feu/Sans Fumée

**Figure 2-1:** Échantillons d'images de trois classes de l'ensemble de données personnalisé

### III.2 Prétraitement des données

Le prétraitement des données est une technique de conversion des données brutes en un ensemble de données propre. Comme les données recueillies proviennent de différentes sources, elles doivent être standardisées et nettoyées avant d'être transmises aux algorithmes d'apprentissage automatique. Le prétraitement est une étape nécessaire pour réduire la complexité et améliorer la précision de l'algorithme. Aucune des opérations de changement de couleur ou d'intensité n'est appliquée à ces données car ces propriétés ont un impact sur le modèle d'apprentissage en temps réel. Changer la couleur et l'intensité lumineuse changera la couleur du feu et de la fumée, ce qui doit être évité car l'algorithme doit les apprendre en tant que caractéristiques principales.

Le nettoyage des données a été effectué pour supprimer les images inutiles qui contiennent trop de détails ou trop peu de détails dans les fonctionnalités. Après nettoyage, les données sont classées en deux classes étiquetées :

Les données ont été extraites de la plateforme kaggle[17]

- Feu
- Fumée
- Normal

#### III.2.1 L'augmentation des données

L'augmentation des données est une technique permettant de créer artificiellement de nouvelles données d'entraînement à partir de données d'entraînement existantes. Cela se fait en appliquant des techniques spécifiques à un domaine à des exemples à partir des données de formation qui créent des exemples de formation nouveaux et différents.

Cela nous aide à augmenter la taille de l'ensemble de données et à introduire de la variabilité dans l'ensemble de données, sans réellement collecter de nouvelles données. Le réseau neuronal traite de toute façon ces images comme des images distinctes. L'augmentation des données aide également à réduire le sur-ajustement. Notre ensemble de données peut avoir des images prises dans un ensemble limité de conditions, mais nous pouvons être en deçà dans une variété de conditions dont nous ne tenons pas compte. Ici, les données modifiées/augmentées aident à faire face à de tels scénarios.

Donc, pour obtenir plus de données, nous devons apporter des modifications mineures à nos données d'entraînement existantes. Ici, nous parlons spécifiquement de l'augmentation des

données d'image. Ces modifications incluent le retournement de l'image horizontalement, verticalement, le rembourrage, le recadrage, la rotation, la mise à l'échelle et quelques autres traductions.[12]

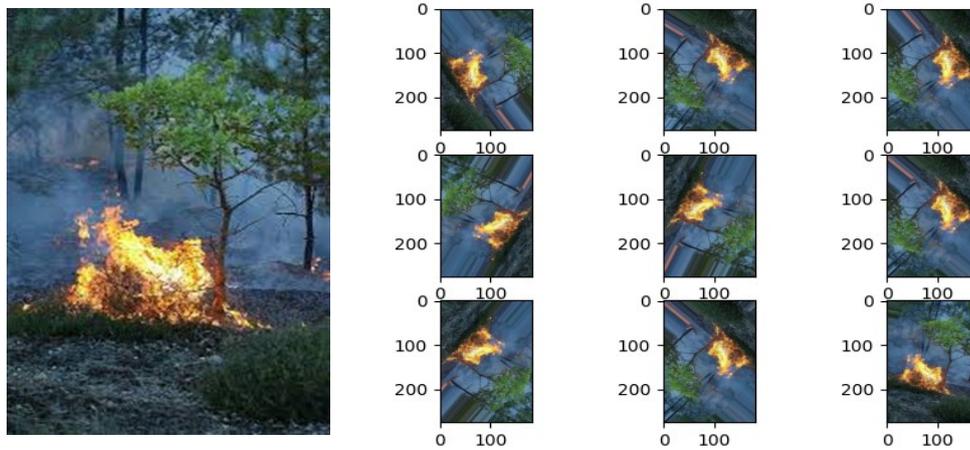


Figure 2-2 : Opérations aléatoires sur un exemple d'image

### III.3 Fractionnement des données

L'une des règles d'or de l'apprentissage automatique consiste à diviser votre ensemble de données en ensemble d'apprentissage, de validation et de test.

La raison pour laquelle nous faisons cela est très simple. Si nous ne divisons pas les données en différents ensembles, le modèle serait évalué sur les mêmes données qu'il a vues pendant la formation. Nous pourrions donc rencontrer des problèmes tels que le sur-ajustement sans même le savoir. Avant d'utiliser des modèles d'apprentissage en profondeur, nous utilisons souvent trois ensembles différents.

- Un train est utilisé pour former le modèle.
- Un ensemble de validation utilisé pour évaluer le modèle pendant le processus de formation.
- Ensemble de test utilisé pour évaluer la précision du modèle final avant le déploiement.

Il existe plusieurs façons de diviser les données en ensembles d'apprentissage et de test. L'approche la plus courante consiste à utiliser un échantillonnage aléatoire. L'échantillonnage aléatoire complet est une stratégie simple à mettre en œuvre et protège généralement le modèle d'un biais en faveur de toute caractéristique des données. Cependant, cette approche peut être problématique lorsque la réponse n'est pas uniformément répartie sur le résultat. Une stratégie de fractionnement moins risquée consisterait à utiliser un échantillon aléatoire stratifié en fonction du résultat. Pour les modèles de classification, cela se fait en sélectionnant des échantillons au

hasard dans chaque classe. Cette approche garantit que la distribution de fréquence du résultat est approximativement égale dans les ensembles d'entraînement et de test [13].

Un ensemble de test séparé a été mis de côté pour vérifier la précision du modèle en temps réel. Ainsi, au lieu de diviser l'ensemble de données existant en ensembles d'apprentissage, de validation et de test, les données ont été divisées en ensembles d'apprentissage et de validation en fonction d'un ratio. Le rapport de division dépend fortement du type de modèle et de l'ensemble de données lui-même. Dans le cas des images et des vidéos, le modèle nécessite beaucoup de formation, puis une plus grande partie des données est utilisée à des fins de formation. Lorsque le modèle a plusieurs hyper paramétrés à régler, il est conseillé de conserver un pourcentage plus élevé de données pour l'ensemble de validation.

### III.3.1 Ensemble d'entraînement

Un ensemble d'apprentissage est une partie d'un ensemble de données utilisé pour ajuster (entraîner) un modèle de prédiction ou de classification de valeurs connues dans l'ensemble d'apprentissage, mais inconnues dans d'autres données (futures). L'ensemble d'apprentissage est utilisé conjointement avec des ensembles de validation et/ou de test qui sont utilisés pour évaluer différents modèles. Les ensembles de formation sont utilisés dans les procédures d'apprentissage supervisé dans l'exploration de données (c'est-à-dire la classification des enregistrements ou la prédiction de valeurs cibles continues.) 80% de l'ensemble de données d'origine est divisé et utilisé comme ensemble de formation.

### III.3.2 Jeu de validation

Un ensemble de validation est utilisé pour valider la manière dont le modèle s'adapte à l'ensemble de données. Il est utilisé pour régler et ajuster les hyper paramètres de l'algorithme (réseau de neurones). Le modèle n'apprendra rien de l'ensemble de validation mais l'utilisera pour ajuster les hyper paramètres afin d'obtenir un modèle généralisé mieux ajusté. Un ensemble de validation fonctionne comme un ensemble hybride utilisé dans la formation à des fins de test, mais pas comme un ensemble de test final. 20% des données

### III.3.3 Ensemble de test avec des scénarios réels

Un jeu de test a été utilisé pour tester l'hypothèse du modèle. Ces données ne sont pas modifiées dans l'entraînement du modèle, même s'il s'agit d'un sous-ensemble de l'ensemble de données d'origine. Habituellement, ces données ne sont jamais vues par le réseau jusqu'à ce qu'il ait terminé la formation et la validation, le réglage des hyper paramètres est finalement défini

pour le modèle. Le modèle final est appliqué sur les données de test pour obtenir une mesure précise de ses performances lorsqu'il est déployé sur des données du monde réel.



Figure 2-3: Une visualisation des divisions[14]

### III.3.4 YOLOv5 Traitement des données

#### Création de jeu de données

Dans ce didacticiel, nous avons créé notre propre ensemble de données sur les pingouins, en marquant manuellement environ 2 000 images. Il nous a fallu quelques heures pour utiliser la plateforme Roboflow, qui est facile à utiliser et gratuite pour les utilisateurs généraux [21].

Pour un modèle YOLOv5 robuste, une formation avec plus de 1 500 images par classe et plus de 10 000 cas par classe est recommandée. Il est également recommandé d'ajouter jusqu'à 10 % d'images d'arrière-plan pour réduire les faux positifs. Étant donné que notre ensemble de données est très petit, nous limiterons le processus de formation en utilisant des techniques d'apprentissage par transfert.

#### Format d'étiquetage YOLO

La plupart des plates-formes d'annotation prennent en charge l'exportation au format d'étiquetage YOLO, fournissant un fichier texte d'annotations par image. Chaque fichier texte contient une annotation de boîte englobant (Box) pour chacun des objets de l'image. Les annotations sont normalisées à la taille de l'image et se situent dans la plage de 0 à 1. Elles sont représentées au format suivant :

< object-class-ID><X center><Y center><Box width><Box height>

0 0.383 0.439 0.183 0.628

0 0.507 0.454 0.191 0.71

# Chapitre III

**Architectures de réseaux neuronaux**

**Introduction**

Dans ce chapitre, nous expliquerons les architectures de réseaux de neurones, définirons les environnements utilisés dans le travail, ainsi que le langage de programmation et enfin nous aborderons le réglage des hyperparamètres

**IV. Architectures de Réseaux Neuronaux**

Les réseaux de neurones artificiels sont un paradigme de programmation d'inspiration biologique qui permet à un ordinateur d'apprendre à partir de données d'observation. Un [ANN] est basé sur une collection d'unités ou de nœuds connectés appelés neurones artificiels, modélisés vaguement d'après le cerveau humain, conçus pour reconnaître des modèles. Les réseaux de neurones peuvent interpréter les données sensorielles par la perception de la machine, l'étiquetage ou le regroupement des entrées brutes. Les modèles qu'ils reconnaissent sont numériques, contenus dans des vecteurs, constitués de données du monde réel comme des images, du son, du texte ou des séries chronologiques, doivent être traduits. Ils sont utilisés pour regrouper, classer, regrouper les données non étiquetées en fonction des similitudes entre les exemples d'entrées et classer les données lorsqu'elles ont un ensemble de données étiqueté sur lequel s'entraînent. Ils peuvent extraire des fonctionnalités qui sont transmises à d'autres algorithmes pour regroupement et classification.

Le chapitre 3 discutera du besoin de réseaux de neurones profonds, d'apprentissage par transfert de capteurs de bas niveau et de leur formation.

**IV.1 Réseaux de neurones profonds (Deep Neural Networks) :**

Le cerveau humain, ses fonctions et son fonctionnement ont inspiré la création du réseau neuronal. L'intelligence artificielle et l'apprentissage automatique, qui est un sous-ensemble de l'IA, jouent un rôle essentiel dans sa fonctionnalité. Il commence à fonctionner lorsqu'un développeur saisit des données et construit un algorithme d'apprentissage automatique, en utilisant principalement le principe ("if ... else ...") de la construction d'un programme. Le réseau neuronal profond ne fonctionne pas seulement selon l'algorithme, mais peut également prédire une solution pour une tâche et tirer des conclusions en utilisant son expérience antérieure. Dans ce cas, vous n'avez pas besoin d'utiliser la programmation ou le codage pour obtenir une réponse.

Un réseau neuronal profond est bénéfique lorsque vous devez remplacer le travail humain par un travail autonome sans compromettre son efficacité. L'utilisation du réseau neuronal profond peut trouver diverses applications dans la vie réelle. Par exemple, une société chinoise **Sensetime**[15] a créé un système de reconnaissance faciale automatique pour identifier les

criminels, qui utilise des caméras en temps réel pour trouver un délinquant dans la foule. De nos jours, c'est devenu une pratique populaire dans la police et d'autres entités gouvernementales.

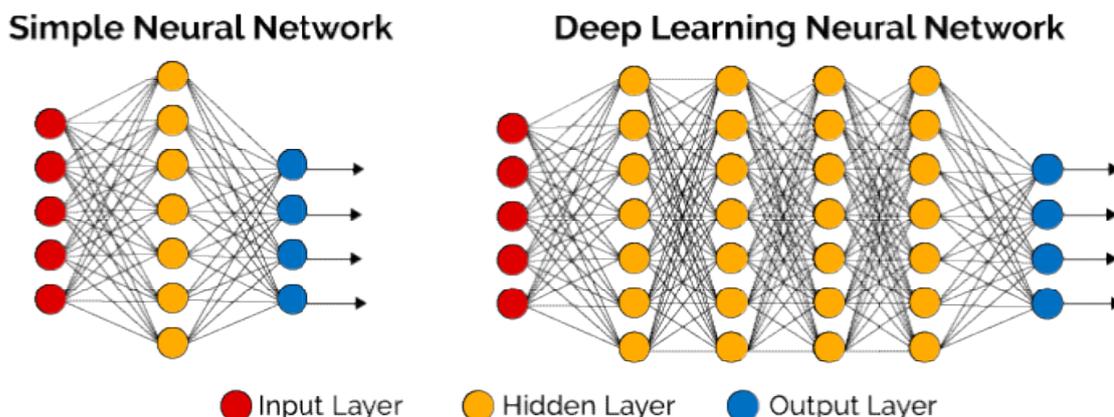


Figure 3-1 : Représentation conceptuelle d'un réseau a Deep neural network [16]

### IV.2 Réseau de neurones convolutif CNN

Il existe différents types de réseaux de neurones et les différences entre eux résident dans leurs principes de travail, le schéma d'actions et les domaines d'application. Les réseaux de neurone convolutif (CNN) sont principalement utilisés pour la reconnaissance d'images et rarement pour la reconnaissance audio. Il est principalement appliqué aux images car il n'est pas nécessaire de vérifier tous les pixels un par un. CNN vérifie une image par blocs, en commençant par le coin supérieur gauche et en se déplaçant pixel par pixel jusqu'à la réussite. Ensuite, le résultat de chaque vérification est transmis à travers une couche convolutif, où les éléments de données ont des connexions tandis que d'autres n'en ont pas. Sur la base de ces données, le système peut produire le résultat des vérifications et peut conclure ce qui est dans l'image.

#### IV.2.1 Image d'entrée

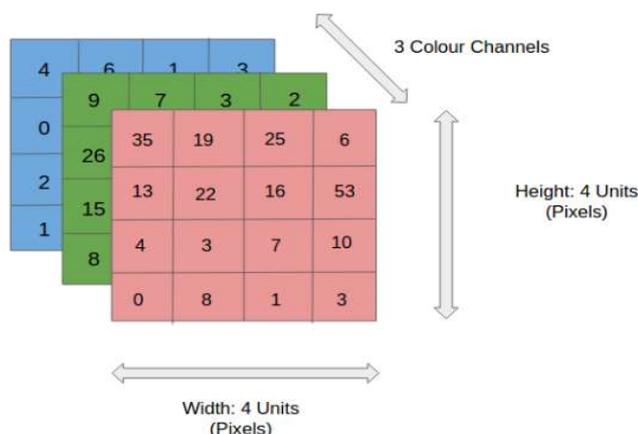


Figure 3-2 :4x4x3 RGB Image [17]

Que toute image couleur a trois canaux, c'est-à-dire le rouge, le vert et le bleu, comme la montre la figure ci-dessus. Il existe plusieurs espaces colorimétriques tels que les niveaux de gris, CMJN, HSV dans lesquels une image peut exister.

IV.2.2 Convolution

La convolution utilise un "noyau" pour extraire certaines "caractéristiques" d'une image d'entrée. Un noyau est une matrice, qui est glissée sur l'image et multipliée par l'entrée de telle sorte que la sortie est améliorée d'une certaine manière souhaitable. Regardez ceci en action ci-dessous

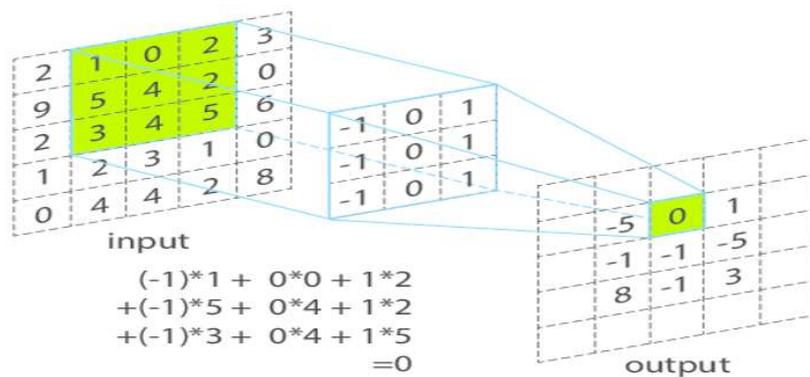


Figure 3-3 : Image illustre comment la convolution est effectuée sur une image d'entrée pour extraire des caractéristiques [18]

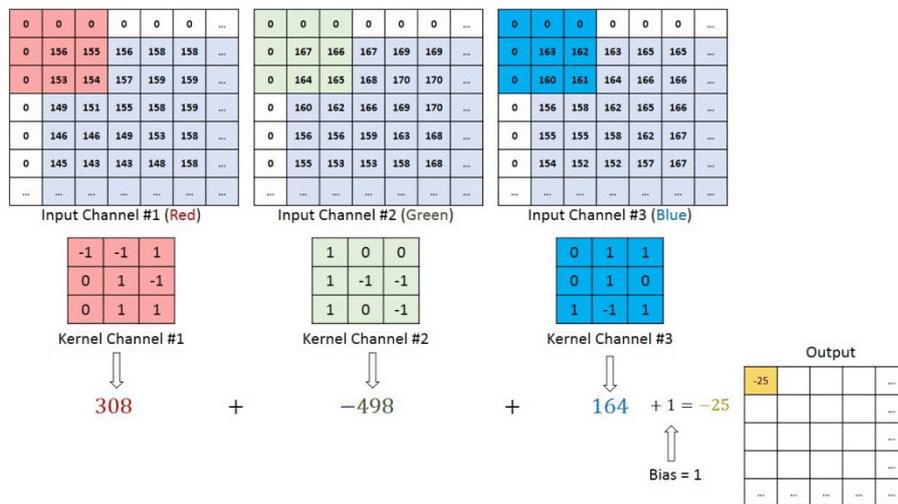


Figure 3-4 : Opération de convolution sur une matrice image MxNx3 avec un Kernel 3x3x3 [17]

### IV.2.3 Couches de pooling (Pooling Layer)

Les couches de regroupement sont utilisées pour réduire les dimensions des cartes d'entités. Ainsi, cela réduit le nombre de paramètres à apprendre et la quantité de calculs effectués dans le réseau.

La couche de regroupement résume les caractéristiques présentes dans une région de la carte de caractéristiques générée par une couche de convolution. Ainsi, d'autres opérations sont effectuées sur les entités résumées au lieu des entités positionnées avec précision générées par la couche de convolution. Cela rend le modèle plus robuste aux variations de la position des caractéristiques dans l'image d'entrée[10].

#### Types de couches de Pooling

**Mise en commun maximale (Max Pooling)** est une opération de mise en commun qui sélectionne le maximum d'éléments dans la région de la carte d'entités couverte par le filtre. Ainsi, la sortie après la couche de regroupement maximal serait une carte d'entités contenant les caractéristiques les plus importantes de la carte d'entités précédente.

**Mise en commun moyenne (Average Pooling)** calcule la moyenne des éléments présents dans la région de la carte d'entités couverte par le filtre. Ainsi, alors que la mise en commun maximale donne la fonctionnalité la plus importante dans un patch particulier de la carte des fonctionnalités, la mise en pool moyenne donne la moyenne des fonctionnalités présentes dans un patch.

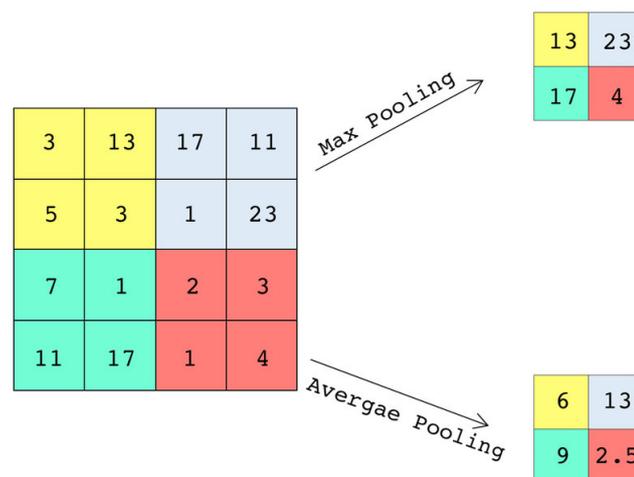


Figure 3-5 : Illustration de la mise en commun maximale et de la mise en commun moyenne [19]

### IV.3 Apprentissage par transfert(Transfer Learning)

L'apprentissage par transfert est un outil important de l'apprentissage automatique pour résoudre le problème fondamental de l'insuffisance des données d'entraînement. Il essaie de transférer les connaissances du domaine source vers le domaine cible en relâchant l'hypothèse selon laquelle les données d'apprentissage et les données de test doivent être distribuées de manière identique. Cela conduit à un grand effet positif sur de nombreux domaines. Le processus d'apprentissage de l'apprentissage par transfert illustré dans la figure.

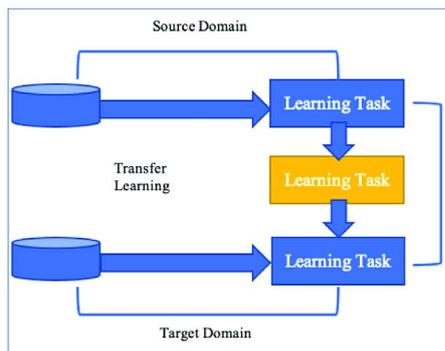


Figure 3-6: Processus d'apprentissage de l'apprentissage par transfert[24]

#### IV.3.1 ImageNet

La principale motivation de l'apprentissage en profondeur est le fait que la plupart des modèles qui résolvent des problèmes complexes ont besoin de beaucoup de données, et obtenir de grandes quantités de données étiquetées pour les modèles supervisés peut être très difficile, compte tenu du temps et des efforts nécessaires pour étiqueter points de données. Un exemple simple serait l'ensemble de données ImageNet, qui contient des millions d'images appartenant à



Figure 8 : Un échantillon de l'ensemble de données ImageNet [20].

différentes catégories.

ImageNet est une base de données d'images destinée aux travaux de recherche dans le domaine de la vision par ordinateur[25].

Cette base de données comporte plus de 15 millions d'images qui appartiennent à 22.000 classes ou catégories étiquetées à la main et recueillies à partir de diverses sources sur le web.

Lancée en 2010, ILSVRC (ImageNet Large-Scale Visual Recognition Challenge) est une compétition annuelle qui a pour but d'obtenir la meilleure prédiction sur plusieurs tâches dans le domaine de la vision par ordinateur tel que la détection et la classification d'objets dans les images naturelles. Cette compétition est basée sur un sous-ensemble d'ImageNet contenant 1,2 million d'images d'entraînement, 50 000 images de validation et 150 000 images de test classées en 1000 catégories. Plusieurs modèles ont été pré-entraînés sur ce sous-ensemble à l'aide de la technique Transfer Learning pour exploiter les connaissances acquises sur celle-ci, et 36 plusieurs recherches ont démontré l'efficacité de pré-entraîner des modèles sur cette base de données. C'est le cas de où les modèles AlexNet, ResNet-50, GoogleNet et VGG16 ont été pré-entraînés[25].

#### **IV.4 Réseaux de neurones pour l'apprentissage par transfert**

La portée de ce projet consiste à trouver le meilleur réseau compatible et convivial sur le plan informatique à déployer sur des capteurs activés par le calcul de bas niveau. Ce réseau doit apprendre les fonctionnalités de l'ensemble de données personnalisé à l'aide de l'apprentissage par transfert sur ImageNet. Les réseaux qui satisfont à toutes ces exigences sont étudiés en détail dans cette section.

##### **IV.4.1 YOLOv5**

YOLOv5 est un détecteur d'objets en temps réel populaire. Il s'agit d'une implémentation PyTorch du détecteur de tir unique YOLO (You Only Look Once) qui est connu pour sa vitesse fulgurante et sa précision raisonnable.

Officiellement, dans le cadre du Framework Darknet, il existe quatre versions de YOLO. De YOLOv1 à YOLOv4. Le Framework Darknet est écrit en C et CUDA.

YOLOv5 est la prochaine version équivalente de la famille YOLO à quelques exceptions près

- Le projet a été lancé par Glenn Jocher sous l'organisation Ultralytics sur Git Hub[21].
- Il a été écrit en langage Python et le Framework utilisé est PyTorch.

##### **Modèles disponibles dans YOLOv5**

Pour commencer à explorer l'ensemble du paysage de YOLOv5, commençons par les modèles. Il contient 5 modèles au total. De YOLOv5 nano (le plus petit et le plus rapide) à YOLOv5 extra-large (le plus grand modèle).

Voici une brève description de chacun d'entre eux :

- YOLOv5n : Il s'agit d'un modèle nano nouvellement introduit, qui est le plus petit de la famille et destiné à la périphérie, aux appareils IoT, et avec le support Open CV DNN également. Il fait moins de 2,5 Mo au format INT8 et environ 4 Mo au format FP32. Il est idéal pour les solutions mobiles.
- YOLOv5s : C'est le petit modèle de la famille avec environ 7,2 millions de paramètres et il est idéal pour exécuter l'inférence sur le CPU.
- YOLOv5m : Il s'agit du modèle de taille moyenne avec 21,2 millions de paramètres. C'est peut-être le modèle le mieux adapté à de nombreux ensembles de données et à la formation, car il offre un bon équilibre entre vitesse et précision.
- YOLOv5l : C'est le grand modèle de la famille YOLOv5 avec 46,5 millions de paramètres. Il est idéal pour les ensembles de données où nous devons détecter des objets plus petits.
- YOLOv5x : C'est le plus grand parmi les cinq modèles et a également le mAP le plus élevé parmi les 5. Bien qu'il soit plus lent que les autres et compte 86,7 millions de paramètres.

Le tableau suivante montre un aperçu encore meilleur de tous les modèles, y compris la vitesse d'inférence sur le CPU, le GPU, ainsi que le nombre de paramètres avec une taille

**Tableau 1** : Le tableau suivante montre un aperçu encore meilleur de tous les Modèles de YOLOV5 [22]

Modèle Nom	Paramètres (Million)	Accuracy (mAP 0.5)	CPU Time (ms)	GPU Time (ms)
YOLOV5n	1.9	45.7	45	6.3
YOLOV5s	7.2	56.8	98	6.4
YOLOV5m	21.2	64.1	224	8.2
YOLOV5l	46.5	67.3	430	10.1
YOLOV5x	86.7	68.9	766	12.1

d'image de 640

L'architecture YOLO est fondamentalement un réseau de neurones à convolution profonde (CNN), comme l'illustre ce diagramme de haut niveau de l'article YOLO original

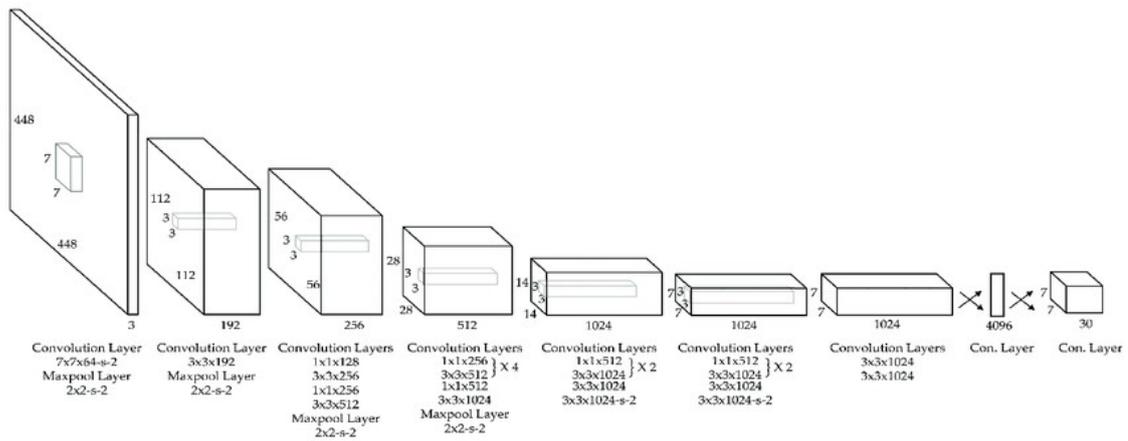


Figure 9 : La structure du réseau du modèle de véhicule YOLOv5 [32]

### IV.4.2 VGG16

VGG16 est un réseau convolutionà 16 couches utilisé par le groupe de géométrie visuelle (VGG) de l'Université d'Oxford dans le cadre du concours ILSVRC (ImageNet) en 2014. Le modèle atteint un taux de succès de 92,5% dans le top 5 sur l'ensemble de validation[28]. L'image suivante représente l'architecture du réseau

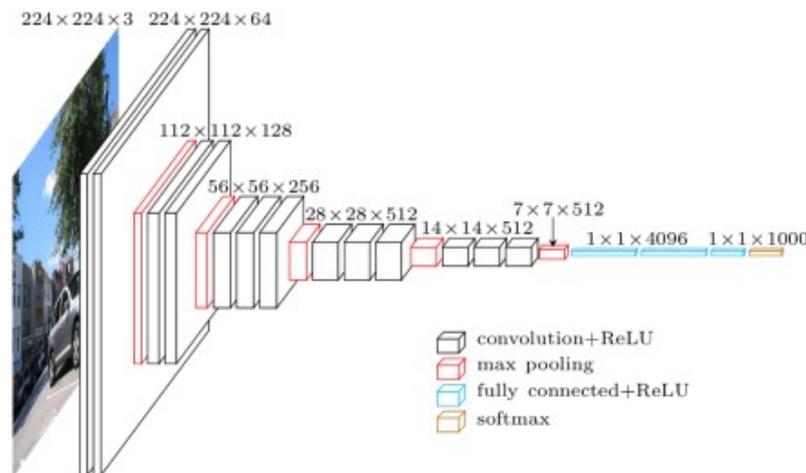


Figure 10 : Représentation 3D de l'architecture de VGG-16 [27]

Il prend en entrée une image en couleurs de taille 224\*224 px et la classifie dans une des 1000 classes. Il renvoie donc un vecteur de taille 1000, qui contient les probabilités d'appartenance à chacune des classes. L'architecture de VGG-16 est illustrée par les schémas ci-dessous[28].

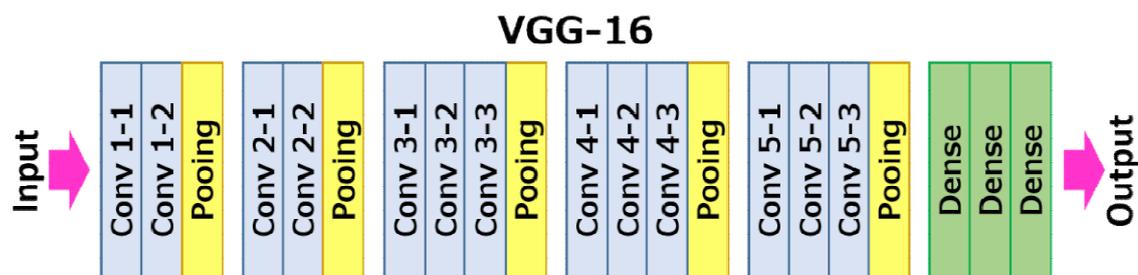


Figure 11 : couches dans l'architecture vgg16[35]

L'extraction automatique de caractéristiques n'exploite que la partie convolutive d'un réseau pré-entraîné. L'utilise comme extracteur de caractéristiques d'image, pour identifier le feu

En pratique, VGG 16 est découpé pour ne conserver que la partie convolutive. Cette partie est appelée gel, pour exprimer l'absence de blocage. Ce réseau prend une image d'entrée au format .Format correct (224, 224, 3) et sortie du code CNN. Ainsi, chaque image de l'ensemble de données est convertie en un vecteur de caractéristiques, qui est utilisé pour former un nouveau classifieur[30].

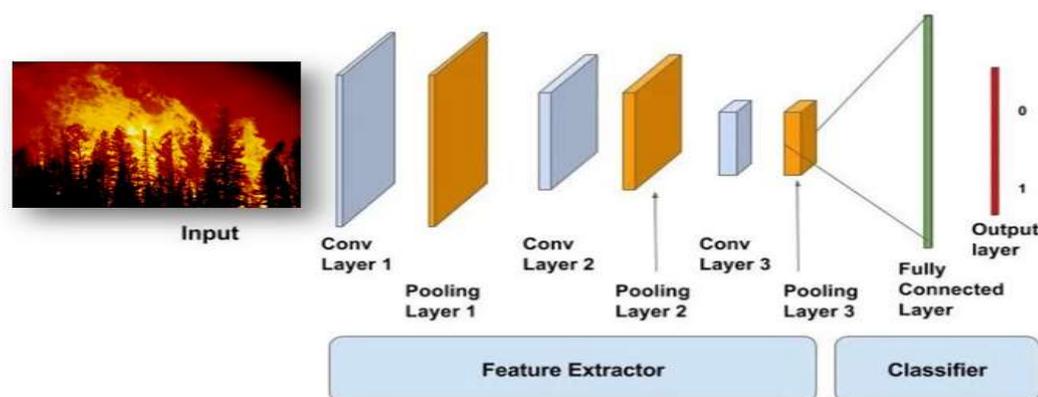


Figure 12 :Représentation de l'extraction automatique de caractéristiques avec VGG-16[30]

## IV.5 VGG19

Le Visual Geometry Group Network (VGGNet) est un réseau neuronal profond multicouche processus. VGGNet est basé sur le modèle CNN et est appliqué à l'ensemble de données ImageNet. VGG-19 est utile en raison de sa simplicité car 3 3 couches convolution elles sont installées sur le dessus pour se développer avec niveau de profondeur. Pour réduire la taille du volume, les couches de regroupement maximales ont été utilisées comme paramètre dans VGG-19. Utilisé comme données d'entrée pour VGGNet. Dans la phase d'apprentissage, des couches convolutives sont utilisées Extraction d'entités et de couches de pooling maximales associées à certaines couches convolutives Réduisez les dimensions des fonctions. Dans la

première couche convolutive, il y avait 64 cœurs (taille de filtre 3\*3). Appliquer pour extraire l'entité des images saisies. Des couches entièrement connectées ont été utilisées pour préparer des vecteurs de fonctions. Le vecteur de caractéristiques acquis est en outre exposé pour PCA et SVD aux dimensions réduites et sélectionnez la fonction de données d'image pour de meilleurs résultats de classification. Réduire haut[31].

Les données dimensionnelles utilisant PCA et SVD sont une tâche importante. PCA et SVD sont plus bénéfiques car elle est plus rapide et numériquement plus stable que les autres techniques de réduction. Technique. Les performances du système basé sur VGG-19 ont été comparées à une autre caractéristique Architectures d'extraction incluant AlexNet et SIFT. AlexNet est une fonction d'extraction multicouche L'architecture utilisée dans CNN. La transformation de caractéristiques fixes à l'échelle (SIFT) est une extraction de caractéristiques classique[31].

Une technique introduite par Mansour pour détecter les caractéristiques locales de l'image d'entrée dans un champ vision par ordinateur.

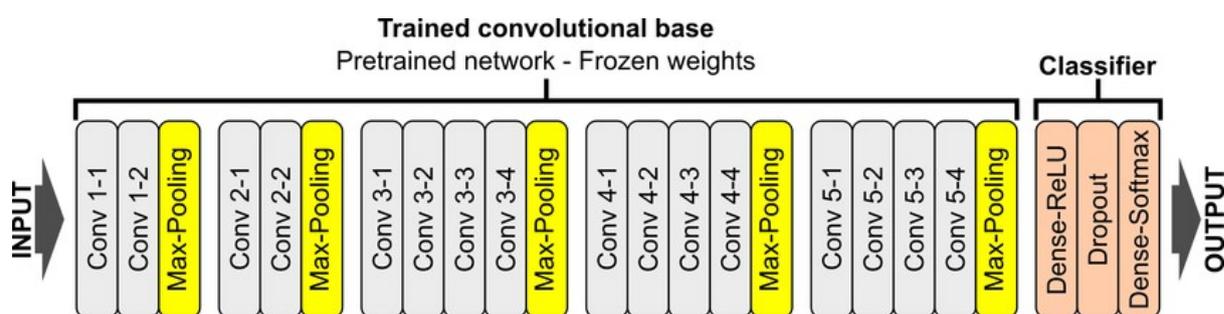


Figure 13 : couches dans l'architecture vgg19 [28]

## IV.6 Réglage des hyperparamètres

Les modèles d'apprentissage automatique sont paramétrés de sorte que leur comportement puisse être ajusté pour un problème donné à partir des données. En formant un modèle avec des données existantes, il sera en mesure d'adapter les paramètres du modèle. Cependant, il existe un autre type de paramètres, appelés hyperparamètres, qui ne peuvent pas être directement appris à partir du processus de formation régulier. Ils sont généralement xés avant le début du processus de formation proprement dit. Ces paramètres expriment des propriétés importantes du modèle telles que sa complexité ou la vitesse à laquelle il doit apprendre.

Un paramètre de modèle est une variable de configuration interne au modèle et dont la valeur peut être estimée à partir des données fournies. Ils sont requis par le modèle pour faire des prédictions. Les valeurs de ces paramètres sont apprises à partir des données et définissent les

performances du modèle sur un problème donné. Ils ne sont souvent pas définis manuellement par le praticien et sont enregistrés dans le cadre du modèle appris. Les paramètres du modèle incluent les pondérations dans un réseau de neurones, les vecteurs de support dans une machine à vecteurs de support et les coefficients de régression linéaire ou logistique.

Un hyperparamètre est un paramètre constant dont la valeur est définie avant le début du processus d'apprentissage et ne peut pas être estimée à partir des données. Les valeurs des paramètres sont dérivées via l'apprentissage, souvent utilisé dans les processus pour aider à estimer les paramètres du modèle. Les valeurs de certains hyperparamètres peuvent être dépendantes les unes des autres. Ces paramètres sont souvent réglés pour un problème de modélisation prédictive donné. Des exemples d'hyperparamètres incluent le taux d'apprentissage, le nombre de couches masquées et la taille du lot.

En général, il n'y a pas de meilleures valeurs connues pour un hyperparamètre de modèle sur un problème donné. Un praticien peut soit utiliser des règles empiriques, copier des valeurs utilisées sur d'autres problèmes, soit rechercher la meilleure valeur par essais et erreurs. Lorsqu'un algorithme d'apprentissage automatique est réglé pour un problème spécifique, vous réglez essentiellement les hyperparamètres du modèle pour découvrir les paramètres du modèle qui aboutissent aux prédictions les plus habiles. Ces paramètres aideront le modèle à apprendre et à converger plus rapidement.

Les sections suivantes discuteront des hyperparamètres utilisés dans ce projet pour des performances optimales sur l'ensemble de données de détection d'incendie et de fumée.

### **Taux d'apprentissage (Learning rate)**

Algorithme. Le modèle peut utiliser soit le taux d'apprentissage  $X_e$ , le taux d'apprentissage décroissant progressivement, les méthodes basées sur l'élan ou les taux d'apprentissage adaptatifs, selon le choix de l'optimiseur tel que SGD, Adam, Adagrad, AdaDelta ou RMSProp pour le réseau de neurones.

### **Nombre d'époques (epochs)**

Le nombre d'époques est le nombre de fois que l'ensemble de l'apprentissage passe par le réseau neuronal. Le modèle devrait continuer à augmenter le nombre d'époques jusqu'à ce qu'il y ait un petit écart entre l'erreur de test et l'erreur d'apprentissage.

### **Taille du lot (Batch size)**

Le mini-batch est généralement préférable dans le processus d'apprentissage d'un réseau convolutif. Une plage de 16 à 128 est un bon choix pour commencer les tests. Habituellement, les CNN sont sensibles à la taille des lots.

Dans cette expérimentation, des pistes ont été conçues pour s'entraîner avec une taille de lot de 16 et jusqu'à 256. Plus la taille du lot est élevée, plus la validation est précise, mais il y a un hic. Avec l'augmentation de la taille du lot, le temps de formation sera plus court et la quantité d'apprentissage sera moindre par rapport à une taille de lot plus petite. Comme la formation a été effectuée avec des GPU, la taille du lot de 64 était suffisante pour former le réseau.

### **Fonction d'activation**

La fonction d'activation introduit une non-linéarité dans le modèle. Les alternatives sont ReLu, sigmoïde, tanh et d'autres fonctions d'activation en fonction de la tâche. Dans cette expérimentation, softmax est utilisé comme fonction d'activation sur des couches entièrement connectées.

### **Optimiseur**

Optimiser met à jour les paramètres de pondération pour minimiser la fonction de perte. La fonction de perte sert de guide au terrain indiquant à l'optimiseur s'il se déplace dans la bonne direction pour atteindre le fond de la vallée, le minimum global. Des exemples d'optimiseurs incluent Adam, Adagrad, Nadam, gradient accéléré Nesterov (NAG), RMSprop et SGD, etc. En général, Adam fonctionne bien dans la plupart des cas.

Adam peut être considéré comme une combinaison de RMSprop et Stochastic Gradient Descent avec momentum. Il utilise les gradients au carré pour mettre à l'échelle le taux d'apprentissage comme RMSprop et il tire parti de l'élan en utilisant la moyenne mobile du gradient au lieu du gradient lui-même comme SGD avec élan. Il s'agit d'une méthode de taux d'apprentissage adaptatif qui calcule les taux d'apprentissage individuels pour différents paramètres. Son nom est dérivé de l'estimation adaptative des moments, car il utilise des estimations des premier et deuxième moments du gradient pour adapter le taux d'apprentissage pour chaque poids du réseau de neurones.

### **Abandonner(Dropout)**

L'abandon est une technique de régularisation préférable pour éviter le sur ajustement (ovretting) dans les réseaux de neurones profonds. La méthode supprime simplement des unités

dans le réseau de neurones en fonction de la probabilité souhaitée. Une valeur par défaut de 0,5 est un bon choix pour tester.

### Réglage fin (Fine-Tuning)

Le réglage fin consiste à effectuer quelques ajustements précis des paramètres pour améliorer encore les performances. Pendant l'apprentissage par transfert, le réglage fin consiste à dégeler quelques-unes ou toutes les couches du modèle pré-formé et à le laisser s'adapter davantage à la tâche à accomplir.

La portée de ce projet implique un apprentissage par transfert et utilise des modèles pré-formés formés sur des classes ImageNet. Il utilise des pondérations pré-formées ImageNet et les ajuste en conséquence pour les caractéristiques de l'ensemble de données sur les incendies et la fumée.

Nous utiliserons une technique courante en apprentissage par transfert : le réglage fin. Comme lamontre la Figure. 23, le réglage fin comprend les quatre étapes suivantes :

1. Pré-entraîner un modèle de réseau neuronal, c'est-à-dire le modèle source, sur un ensemble de données source (par exemple, l'ensemble de données ImageNet).
2. Créez un nouveau modèle de réseau de neurones, c'est-à-dire le modèle cible. Cela copie toutes les conceptions de modèle et leurs paramètres sur le modèle source, à l'exception de la couche de sortie. Nous supposons que ces paramètres de modèle contiennent les connaissances tirées de l'ensemble de données source et que ces connaissances seront également applicables à l'ensemble de données cible. Nous supposons également que la couche de sortie du modèle source est étroitement liée aux étiquettes de l'ensemble de données source ; il n'est donc pas utilisé dans le modèle cible.
3. Ajoutez une couche de sortie au modèle cible, dont le nombre de sorties correspond au nombre de catégories dans le jeu de données cible. Initialisez ensuite aléatoirement les paramètres du modèle de cette couche.
4. Entraînez le modèle cible sur le jeu de données cible, tel qu'un jeu de données de chaise. La couche de sortie sera formée à partir de zéro, tandis que les paramètres de toutes les autres couches sont affinés en fonction des paramètres du modèle source.

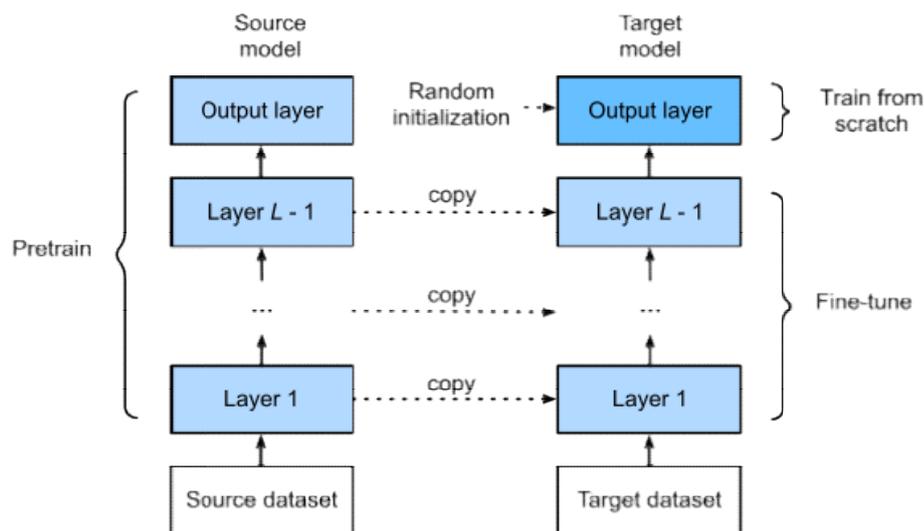


Figure 14 : Fine tuning. [23]

#### IV.7 Implémentation du réseau

Une fois après avoir sélectionné les données, l'architecture réseau et l'ensemble d'hyperparamètres appropriés pour le réseau choisi, il est temps de commencer la formation sur le réseau final. La section suivante traitera de l'implémentation, des bibliothèques de support et du matériel utilisé pour l'expérimentation de ce projet.

##### Base de données

Comme mentionné précédemment dans le chapitre 2, l'ensemble de données était composé de deux classes (Feu, Fumée, Normal) et divisé en 80 % d'ensembles de train et 20 % de validation.

##### Outils et bibliothèques

L'ensemble de l'expérience a été mis en œuvre à l'aide du langage Python et de Keras, une bibliothèque de réseau de neurones open source qui s'exécute au-dessus de TensorFlow. Les modèles et poids pré-formés peuvent être appelés avec une seule commande de ligne à l'aide de la bibliothèque Keras et prennent en charge la formation multi-GPU.

Le réseau a été formé à l'aide des services Google Colab, les spécifications données par Google Colab :

- CPU: Model name: Intel(R) Xeon(R) CPU @ 2.30GHz
- Address sizes: 46 bits physical, 48 bits virtual

- Cache size: 46080 KB
- GPU: single 12GB NVIDIA Tesla K80 GPU
- It can be used up to 12 hours continuously
- 13 GB RAM
- Cloud TPU with 180 teraflops of computation, Intel Xeon processor with two core @ 2.30 GHz
- 13 GB RAM

L'un des inconvénients de Google Colab est qu'il ne fournit que 12 heures de travail

Cette expérience étend la mise en œuvre du réseau à VGG16, VGG19 et YOLOv5 pour trouver une comparaison des performances entre ces réseaux. Ces comparaisons seront discutées dans le chapitre suivant.

# Chapitre IV

Résultats expérimentaux

## V. Résultats expérimentaux

### Introduction

Après avoir effectué l'ingénierie des caractéristiques, sélectionné l'ensemble de données et implémenté le modèle résultant en une sortie sous la forme d'une probabilité ou d'une classe, l'étape suivante consistait à déterminer l'efficacité du modèle basé sur une métrique à l'aide d'ensembles de données de test. L'évaluation de l'algorithme d'apprentissage automatique est une partie essentielle de tout projet.

Le chapitre 4 présente les mesures de performance utilisées pour évaluer l'expérience et les résultats obtenus grâce aux expériences sur les réseaux discutés dans le chapitre précédent.

### V.1 Indicateurs de performance

Bien que la préparation des données et la formation d'un modèle d'apprentissage automatique soient une étape clé du pipeline d'apprentissage automatique, il est tout aussi important de mesurer les performances de ce modèle formé. Il est important de savoir dans quelle mesure le modèle généralise les données invisibles et si le modèle peut être adopté pour résoudre le problème.

Les métriques d'évaluation sont liées aux tâches d'apprentissage automatique. Il existe différentes métriques pour différentes tâches. Cette expérimentation est basée sur la classification, donc l'accent sera mis sur les métriques qui peuvent évaluer une tâche de classification.

**Matrice de confusion** Une matrice de confusion est une matrice  $N \times N$ , où  $N$  est le nombre de classes prédites. Pour le problème en cours, il y a 3 classes et une matrice  $3 \times 3$ . Voici les valeurs représentées dans la matrice de confusion :

- **Vrai positif (TP)** - Un résultat où le modèle prédit correctement la classe positive.
- **Vrai négatif (TN)** - Un résultat où le modèle prédit correctement la classe négative.
- **Faux positif (FP)** - Un résultat où le modèle prédit de manière incorrecte la classe positive.
- **Faux négatif (FN)** - Un résultat où le modèle prédit de manière incorrecte la classe négative.

Tableau 2 : Matrice de confusion et formules

Confusion Matrix		Vrai/Réel			
		Vrai	Faux		
Modèle prédictions	Vrai	TP	FP	<b>Précision</b>	= TP/(TP+FP)
	Faux	FN	TN	NégativePrédictive Value	=TN/ (FN+TN)
		Sensibilité/rappel = TP/ (TP+FN)	Spécificité =TN/ (FP+TN)	Accuracy = (TP+TN)/(TP+FP+TN+FN)	

Les valeurs suivantes peuvent être tirées à l'aide de la matrice de confusion pour évaluer les performances d'un modèle :

**Accuracy** : La proportion du nombre total de prédictions qui étaient correctes.

**Précision** : La proportion de cas positifs qui ont été correctement identifiés.

**Sensibilité ou rappel** : La proportion de cas positifs réels qui sont correctement identifiés.

**Spécificité** : La proportion de cas négatifs réels qui sont correctement identifiés.

**F1-Score** : la moyenne pondérée ou la moyenne harmonique de la précision et du rappel. Ce score tient compte à la fois des faux positifs et des faux négatifs. Le score F1 est un nombre compris entre 0 et 1

**Métriques de classification multi-classes**

**La précision macro** est le rapport entre la quantité de prédictions correctes. Il fait la moyenne sur le nombre d'étiquettes dans la classification. Le numérateur trouve combien d'étiquettes dans le vecteur prédit ont en commun avec la vérité terrain, et le rapport calcule combien d'étiquettes vraies prédites sont réellement dans la vérité terrain.

$$\text{précision macro} = \frac{TP}{TP + FP}$$

**Le rappel de macro** est le rapport du nombre d'étiquettes réelles qui ont été prédites. Le numérateur trouve combien d'étiquettes dans le vecteur prédit ont en commun avec la vérité terrain, puis trouve le rapport au nombre d'étiquettes réelles, obtenant ainsi quelle fraction des étiquettes réelles a été prédite.

$$\text{Recall} = \frac{TP}{TP + FN}$$

**Macro F1-Score** : La macro-moyenne F-mesure est la variante F-measure largement préférée dans les paramètres multi classes en raison de son accent égal sur les classes rares. Il étend la mesure F aux paramètres multi classes en utilisant des valeurs de précision et de rappel moyennées sur toutes les classes

Il traduit l'équilibre entre la précision et le rappel. En termes simples, nous pouvons dire que c'est en fait la moyenne harmonique des deux.

$$2*((\text{précision}*\text{rappel})/ (\text{précision}+\text{rappel})) \quad (7)$$

$$F1 = 2. \frac{\text{PRE. REC}}{\text{PRE} + \text{REC}}$$

## V.2 Comparaisons de performances entre les réseaux

Cette expérimentation a mené une analyse préliminaire sur les performances d'entraînement des réseaux pour déterminer le réseau final à traiter pour le problème de détection. L'analyse préliminaire a été menée sur les réseaux VGG16, VGG19, et YOLOv5.

Performances testées par rapport à un micro-ensemble de 651 images pour comparer l'efficacité du réseau

### V.2.1 Performances de VGG16

VGG16 a été formé à l'apprentissage par transfert pendant 100 époques et à régler l'autre moitié des calques. En déverrouillant ces couches, les pondérations correspondantes ont été apprises et ajustées en fonction des caractéristiques de l'ensemble de données personnalisé. Le tableau

montre  
obtenue après  
de  
sur les

Layer (type)	Output Shape	Param #
vgg16 (Functional)	(None, 7, 7, 512)	14714688
flatten (Flatten)	(None, 25088)	0
dense (Dense)	(None, 50)	1254450
dense_1 (Dense)	(None, 20)	1020
dense_2 (Dense)	(None, 3)	63
Total params : 15, 970,221 Trainable params : 1, 255,533 Non-trainable params : 14, 714,688		

**Tableau 3** : Couches dans la nouvelle architecture du modèle

suivant  
l'architecture  
le processus  
modification  
couches

Les paramètres d'entraînement ont également été sélectionnés comme suit

Loss='categorical\_crossentropy', Optimizer='Adam', Epochs=100, Batch size=30

Après entraînement sur les paramètres ci-dessus pendant 3 heures, les courbes suivantes ont été obtenues

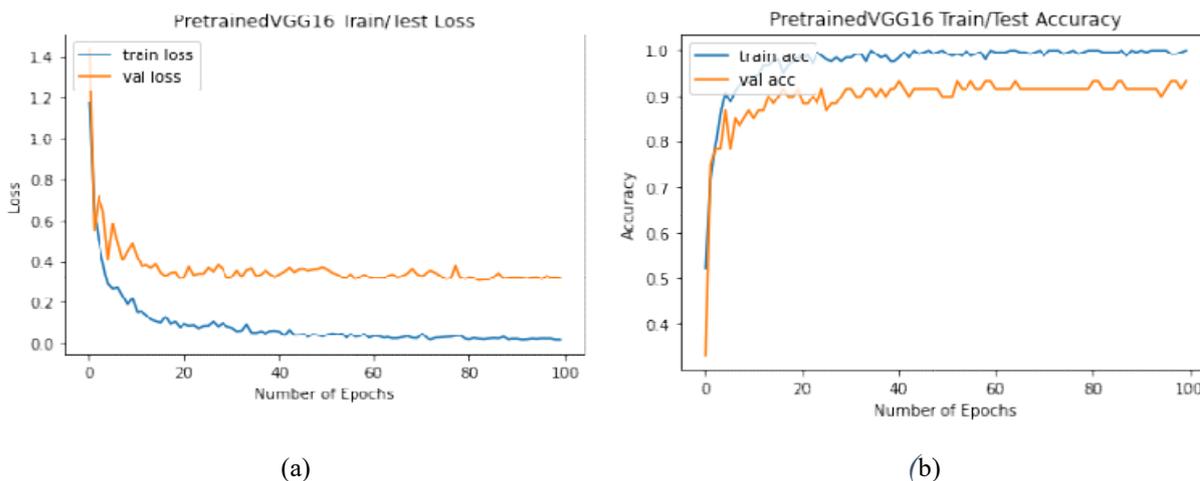


Figure 4-1 : Graphiques d'entraînement de VGG16

Le réseau a atteint sa précision d'entraînement maximale de 98,38 % à la fin de 26 époques avec une précision de validation de 96,67 %. Le temps de formation pour VGG16 a été estimé à 3 heures 10 minutes. Le modèle formé a été évalué par rapport au même mini-ensemble de tests et a obtenu une précision de 32 % et un score Macro f1 de 0,31 et un recall de 0,32.

Tableau 4 : Matrice de confusion de VGG16 sur mini-testset

Confusion Matrix VGG16		Vrai/Réel		
		Feu	Non-feu/Non-fumée	Fumée
Modèle prédictions	Feu	105	84	122
	Non-feu/Non-fumée	66	47	67
	Fumée	49	44	67

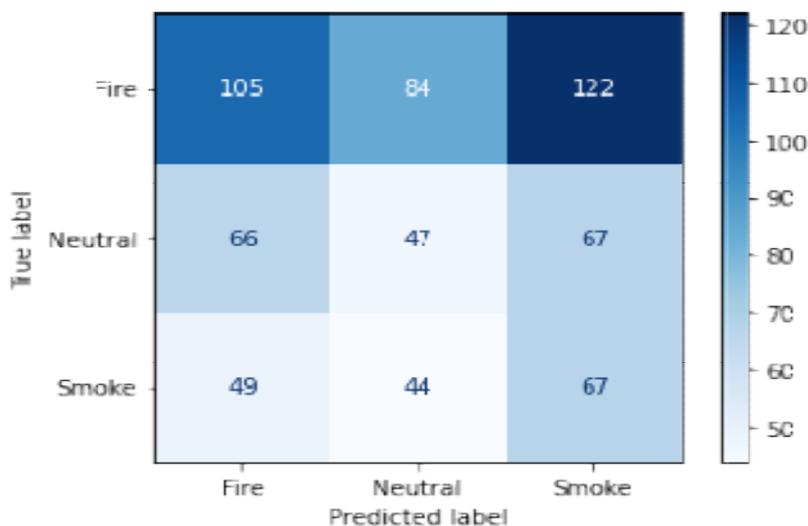


Figure 4-2 : Matrice de confusion de VGG16 sur mini-testset

### V.2.2 Performances de VGG19

VGG19 a été formé à l'apprentissage par transfert pendant 100 époques et à régler l'autre moitié des calques. En déverrouillant ces couches, les pondérations correspondantes ont été apprises et ajustées en fonction des caractéristiques de l'ensemble de données personnalisé. Le tableau suivant montre l'architecture obtenue après le processus de modification sur les couches

Layer (type)	Output Shape	Param #
vgg19 (Functional)	(None, 7, 7, 512)	20024384
flatten_1 (Flatten)	(None, 25088)	0
dense_3 (Dense)	(None, 50)	1254450
dense_4 (Dense)	(None, 20)	1020
dense_5 (Dense)	(None, 3)	63
Total params : 21, 279,917 Trainable params : 1, 255,533 Non-trainable params : 20, 024,384		
<b>Tableau 5:</b> Couches dans la nouvelle architecture du modèle		

VGG19 a été formé avec les mêmes paramètres d'hyperparamètres et époques que VGG16. Les graphiques obtenus au cours de la formation sont présentés dans les figures suivantes

Le réseau a atteint sa précision d'entraînement maximale de 97,38 % à la fin de 26 époques avec une précision de validation de 70,12 %. Le temps de formation pour VGG19 a été estimé à 3 heures 10 minutes. Le modèle entraîné a été évalué par rapport au même mini-ensemble de tests et a obtenu une précision de 32 % et un rappelé (recall) de 0,34 et Macro f1-score de 0,31.

Tableau 6 : Matrice de confusion de VGG19 sur mini-testset

Confusion Matrix VGG19		Vrai/Réel		
		Feu	Non-feu/Non-fumée	Fumée
Modèle prédictions	Feu	81	108	122
	Non-feu/Non-fumée	57	68	55
	Fumée	40	68	52

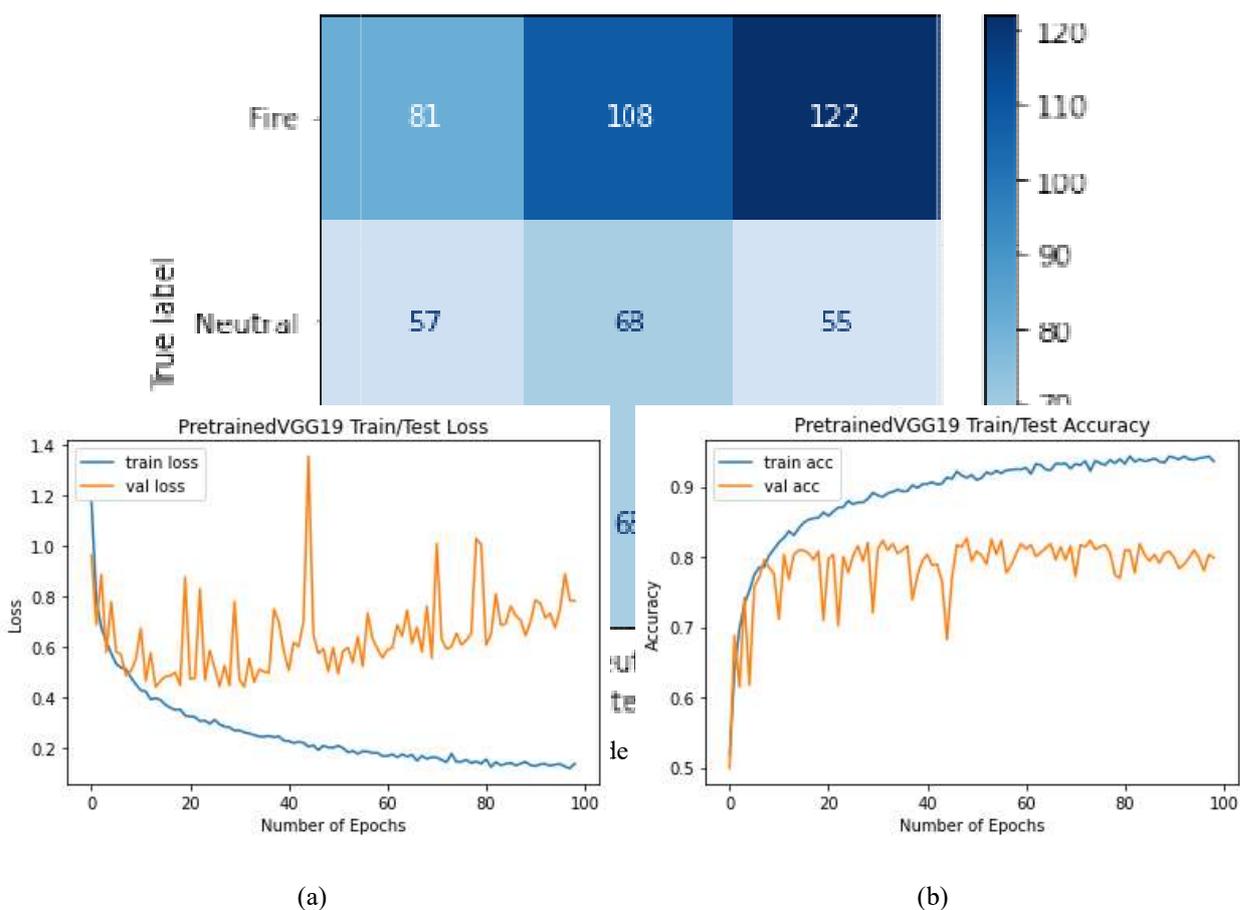


Figure 4-3 : Graphiques d'entraînement de VGG19

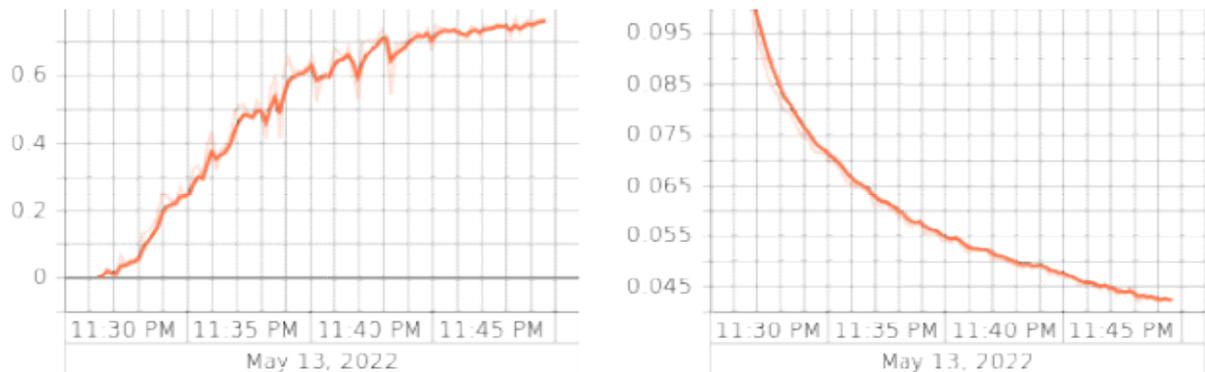


### V.2.3 Performances de YOLOv5

La forme des résultats obtenus à partir de la formation yolov5 est différente car elle est basée sur le TensorBoard, qui produit tous les résultats en même temps.

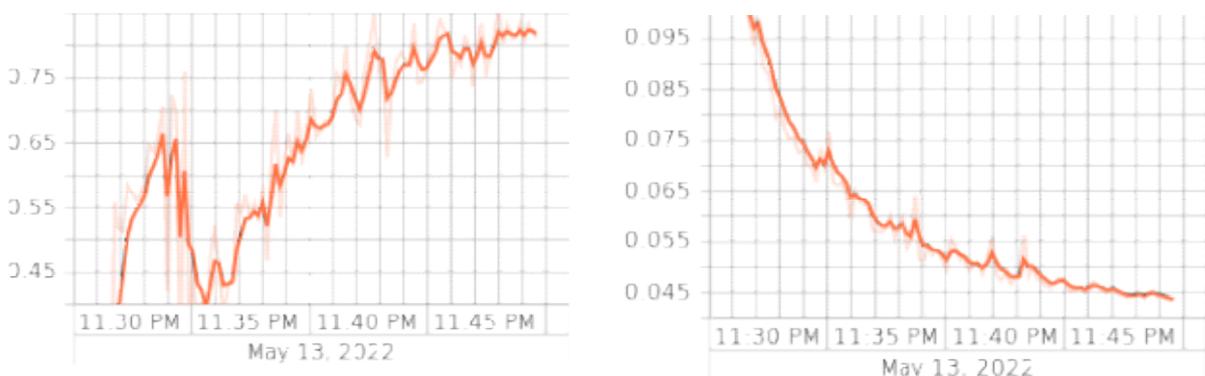
YOLOv5 a été formé sur des paramètres quelque peu différents des réseaux précédents car son architecture l'exige. La taille de l'image a dû être changée en 416\*416.

100 époques ont été formées. À chaque époque, 16 images sont extraites des données d'un



a) Training accuracy

b) Training loss



c) Validation accuracy

d) Validation loss

**Figure 4-5 :** Graphiques d'entraînement de YOLOv5

lot. Le graphique obtenu sur elle par La formation est présentée dans les figures suivantes.

Le réseau a été formé sur 100 époques et a donné une précision de formation de 76,08 % et une précision de validation de 81.02% Le temps de formation sur yolov5 était d'environ deux heures.

Le modèle formé a été évalué par rapport à un petit ensemble de tests et avait une précision de 83,51 % et un score Macro f1 de 0.75, rappeler (recall) de 0.682

Tableau 7 : Matrice de confusion de VGG19 sur mini-testset

Confusion Matrix YOLOv5		Vrai/Réel		
		Feu	Fumée	Background FP
Modèle prédictions	Feu	0.82		0.47
	Fumée		0.77	0.53
	Background FN	0.18	0.23	

L'image suivante représente la matrice de confusion pendant la phase de test

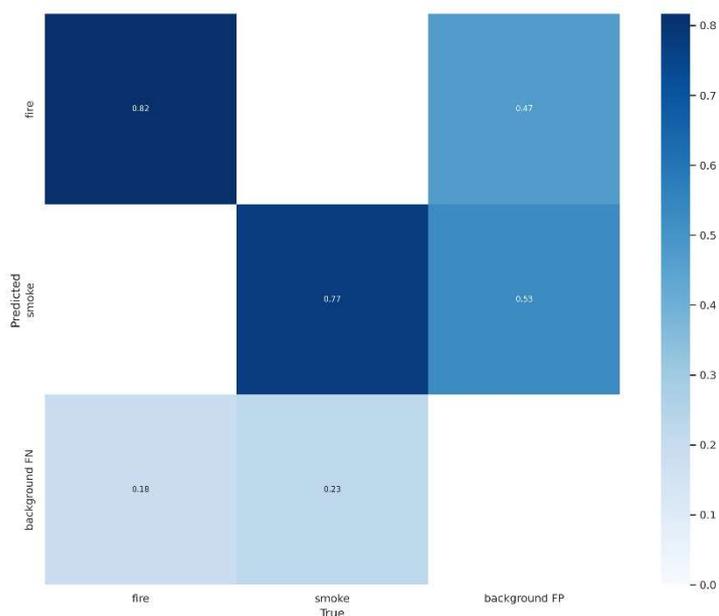


Figure4-6 : Matrice de confusion de VGG19

Les résultats obtenus pour le Macro-précision, le Rappel de macro et le Macro F1-Score sont résumés pour tous les réseaux afin que nous puissions comparer avec une image plus précise. Les résultats sont répertoriés dans le tableau suivant.

Tableau 8 : Précision, rappel et F1-Scores de différents réseaux sur mini-testset

	Macro-précision	Rappel de macro	Macro F1-Score
VGG16	32%	0.32	0.31
VGG19	32%	0.34	0.31
YOLOv5	83.51%	0.6827	0.75

Les images qui ont été testées sur le réseau yolov5, et le feu et la fumée y ont été découverts, et un carré a été placé, et le nom de l'objet (feu ou fumée)

Pour ce qui est de. Le troisième objet représenté dans le normal, il n'a pas été inclus dans le réseau yolov5 car il est difficile de mettre un cadre sur l'objet du fait de sa grande diversité

À l'avenir, la modification sera apportée au code source afin que le modèle puisse fonctionner et détecter s'il n'y a pas de feu ou de fumée, le problème sera résolu par

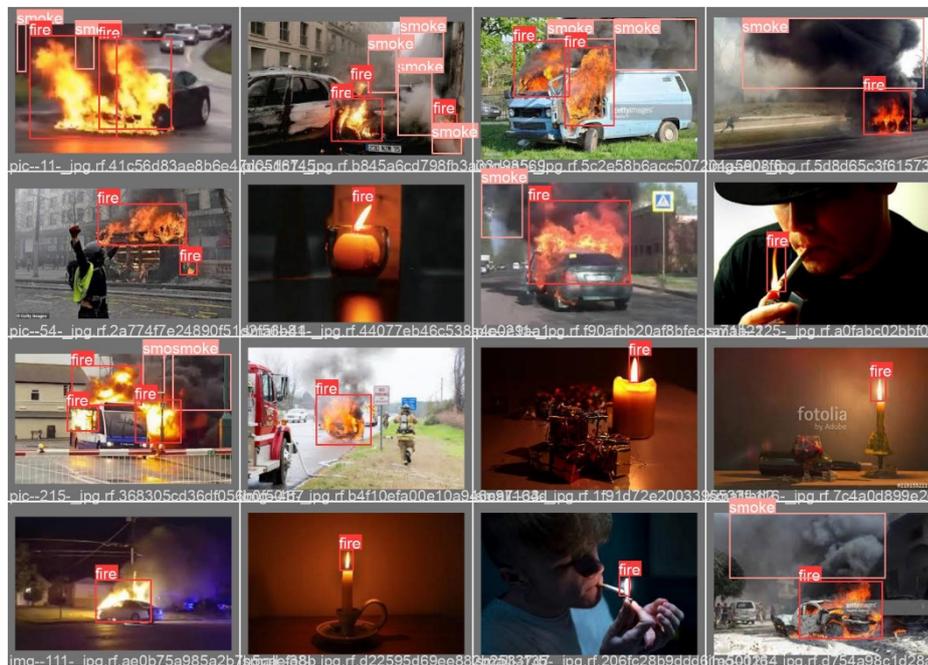


Figure 4-7: Images testées sur le réseau yolov5 .

programmation

### V.3 Comparaison

Parmi les résultats précédents, nous avons observé une augmentation de la valeur de précision sur les réseaux vgg16 et vgg19, ainsi qu'une diminution de la précision sur le réseau yolov5

Mais en comparant les résultats obtenus à partir du test de réseau, nous notons une diminution significative de la note de VGG Catégorie , qui atteint 32%, mais au réseau yolov5, les performances étaient excellentes, atteignant 83.08% et les résultats des tests étaient bons par rapport à VGG Catégorie donc le l'accent est mis sur le réseau yolov5 à l'avenir et travaillez dessus pour améliorer les performances car il convient aux appareils faibles contrairement à celui de la classe VGG qui nécessite un matériel de haut niveau pour fonctionner

### **VI. Conclusion Générale et travaux futurs**

#### **Conclusion**

Les systèmes de détection d'incendie et de fumée sont essentiels pour sauver des vies et prévenir les dangers avant qu'ils ne deviennent incontrôlables. Les technologies existantes de détection d'incendie et de fumée ne peuvent pas jouer efficacement leur rôle dans des endroits où l'espace et les environnements extérieurs sont abondants. L'intelligence artificielle, l'apprentissage automatique et l'apprentissage en profondeur ont donné naissance à de nouvelles approches pour résoudre certains problèmes difficiles à résoudre en utilisant des approches traditionnelles.

Basée sur la vision par ordinateur, cette thèse utilise les réseaux de neurones profonds qui sont compatibles avec les calculs pour les appareils mobiles et les capteurs intelligents pour détecter le feu et la fumée en temps réel à travers les caméras de surveillance installées dans les environnements extérieurs. Il utilise les dernières tendances de l'apprentissage en profondeur grâce à l'apprentissage par transfert sur les réseaux de neurones. Cette thèse utilise un ensemble de données personnalisé de 6 396 images avec plus de 2 100 images dans chaque classe et résout le problème de détection d'incendie et de fumée

#### **Travail futur**

Les futures implémentations peuvent étendre l'expérimentation pour trouver la région d'intérêt (RoI) dans les images d'incendie et de fumée détectées à partir de la vidéo de surveillance en temps réel. Les travaux futurs peuvent étendre la mise en œuvre et les algorithmes à d'autres appareils concurrents de pointe d'autres concurrents tels que Google, Nvidia et Advantech. Google a introduit un accélérateur USB compatible Tensor Processing Unit (TPU) pour accélérer les calculs de périphérie et réduire les temps d'inférence sur les appareils. Avec leur dernière mise à jour, y compris la prise en charge de Windows10 et macOSs'améliore

### VII. Références

- [1] M. A. a. B. Evarts, «Fire loss in the united states during 2020,» NFPA, September 2021.
- [2] J. Selig, «What Is Machine Learning? A Definition.,» 14 March 2022.
- [3] U. M. ., a. J. S. Dan Cireşan, «Multi-column Deep Neural Networks for Image Classification,» Submitted on 13 Feb 2012.
- [4] J. Wu, «AI, Machine Learning, Deep Learning Explained Simply,» Jul 1, 2019.
- [5] S. C. B. S. B. Gardiner, «A Design Procedure for Gradient Operators on Hexagonal Images,» October 2007.
- [6] G. H. G. P. K. Diyana Kyuchukova, «An artificial intelligence approach to real-time automatic smoke detection by unmanned aerial vehicles and forest observation systems,» September 2020.
- [7] R. A. o. P. Smoke, «Ionization Smoke Detector and a Fire Alarm Control Panel with Both Detectors AS,» chez *International Journal of Engineering and Advanced Technology (IJEAT)*, December, 2019.
- [8] T. G. Cleary, «Improving Smoke Alarm\_ Justification for New Smoldering and Flaming Test Performance Criteria,» chez *NIST Technical Note 1837* .
- [9] «Wiring Diagram For Smoke Alarms FULL,» <https://www.happysplanet.top/>.
- [10] «CNN | Introduction to Pooling Layer,» geeksforgeeks, 29 Jul, 2021.
- [11] «What Is a Photoelectric Smoke Detector,» <https://www.x-sense.com>.
- [12] L. Kaiser, «What is an air sampling smoke detection system? Blog,» Fire protection experts, 07 2015.
- [13] W. F. Safety, «rail air sampling smoke detector,» <https://rail.wagnergroup.com/en/system-solutions/fire-detection/air-sampling-smoke-system.html>.
- [14] Y. L. M. R. A. Rui Chen, «Forest Fire Detection Algorithm Based on Digital Image,» *Journal of Software* , 8(8), aug 2013..
- [15] F. C. ., Z. Jing Yang, «Visual-Based Smoke Detection Using Support Vector Machine,» October 2008.
- [16] S. M. P. Jayavardhana Gubbi, «Smoke detection in video using wavelets and support vector machines,» *Fire Safety Journal*, pp. Pages 1110-1115, November 2009.
- [17] X. C. M. N. a. W. Q. Y. Dongqing Shen, «In 2018 4th International Conference on Control, Automation and Robotics (ICCAR),,» Dongqing Shen, Xin Chen, Minh Nguyen, and Wei Qi Yan., IEEE, apr 2018..

## Références

---

- [18] D. L. ahmed saied, «Fire Dataset ,Fire and Smoke Dataset,»  
<https://www.kaggle.com/datasets/phylake1337/fire-dataset>  
,<https://www.kaggle.com/datasets/dataclusterlabs/fire-and-smoke-dataset>.
- [19] A. Gondhalekar, «Data Augmentation — Is it really necessary,» Mar 24, 2020.
- [20] M. K. a. K. Johnson, «Feature Engineering and Selection,» 2019-06-21 .
- [21] T. Shah, «About Train, Validation and Test Sets in Machine Learning,» Dec 6, 2017.
- [22] « <https://roboflow.com/>».
- [23] sensetime, «<https://www.sensetime.com/en>».
- [24] F. Vázquez, «Deep Learning made easy with Deep Cognition,» Dec 21, 2017.
- [25] S. Saha, «A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way,» Dec 15, 2018.
- [26] Convolution, «<https://perso.esiee.fr/~perretb/I5FM/TAI/convolution/index.html>».
- [27] N. Aljaafari, Ichthyoplankton Classification Tool using Generative Adversarial Networks and Transfer Learning, King Abdullah University of Science and Technology, February 2018.
- [28] «Deep Learning Based Real-Time Body Condition Score Classification System,» Kerim Kürşat Çevik , Akdeniz University, November 2020IEEE Access 8:213950-213957.
- [29] «<https://cs.stanford.edu/people/karpathy/cnnembed/>».
- [30] I. S. e. G. E. H. A-Krizhevsky, «mageNet classification with deep convolutional neural networks,» In Advances in Neural Information Processing Systems , 25, NIPS 2012.
- [31] «<https://github.com/ultralytics>,» Ultralytics.
- [32] S. Rath, «Custom Object Detection Training using YOLOv5,» April 19, 2022 .
- [33] A. K. ., M. A. ., A. S. Adel Ammar, «Aerial Images Processing for Car Detection using Convolutional Neural Networks: Comparison between Faster R-CNN and YoloV3,» October 2019.
- [34] A. Ikram, «Une approche IA pour la reconnaissance des expressions faciales,» Universit Akli Mohand Oulhadj de Bouira, 2020.
- [35] R. Thakur, «Step by step VGG16 implementation in Keras for beginners,» Aug 6, 2019.
- [36] «Face recognition using Transfer learning and VGG16,» 2/ 2020.
- [37] K. Bouabdallaoui, «Détecter, identifier et localiser des oiseaux ravageurs des culturesn,» Université de Franche-Comté , Août 2019.

## Références

---

- [38] M. Mateen, «Fundus Image Classification Using VGG-19 Architecture with PCA and SVD,» Chongqing University, 22 November 2018.
- [39] P. K. ., G. Joanna Jaworek-Korjakowska, «Melanoma Thickness Prediction Based on Convolutional Neural Network With VGG-19 Model Transfer Learning,» June 2019.
- [40] Z. C. L. ., L. ., A. J. S. Aston Zhang, «Dive into Deep Learning ,Computer Vision».
- [41] What Is Training Data in Machine Learning?, Rachel Wolff, November, 2020.
- [42] R. Parmar, «Common Loss functions in machine learning,» Sep 2, 2018.