



Mémoire de fin d'étude

Pour l'obtention du diplôme d'ingénieur

Filière : Automatique  
Spécialité : Automatique

Présenté par : BARAKA Hodhaifa Abdelghani

Thème

**Optimisation d'un système automatique  
de détection du covid 19**

Soutenu publiquement, le 27 / 10 / 2020, devant le jury composé de :

M. ABDELLAOUI Ghouti	MCB	ESSA.Tlemcen	Président
M. RIMOUCHE Ali	MCB	ESSA.Tlemcen	Directeur de mémoire
Mme. HANDOUZI Wahida	MCB	Université de Tlemcen	Co- Directrice de mémoire
Mme. GHOMRI Latéfa	MCA	Université de Tlemcen	Examineur 1
Mme. SEBBAGH Hafidha	MCB	ESSA.Tlemcen	Examineur 2

Année universitaire : 2019/2020

*Je dédie ce modeste travail :*

*A mes chers parents*

.

.

.

*Hodhaifa*

## **REMERCIEMENT :**

*Avant toute personne, je remercie le bon Dieu de nous avoir prêté vie, santé et volonté pour achever ce modeste travail.*

*Je tiens à remercier mes encadrants M. RIMOUCHE Ali et Mme. HANDOUZI Wahida pour tout le temps qu'ils m'ont consacré, pour leurs conseils précieux, pour toute leur aide et leur appui durant la réalisation de ce travail.*

*Je tiens à remercier, mes chers parents pour leur encouragement et soutien.*

*Je tiens à remercier chacun des membres du jury pour nous avoir fait l'honneur d'examiner et d'évaluer notre travail*

**Résumé :** L'objectif de ce travail est de concevoir un système de détection du coronavirus, La pandémie de COVID-19 est une pandémie d'une maladie infectieuse émergente, appelée la maladie à coronavirus 2019 ou COVID-19, provoquée par la coronavirus SARS-CoV-2, apparue à Wuhan en chine le 17 novembre 2019, avant de se propager dans le monde à très grande échelle ; d'où la nécessité de mettre en place un système de détection automatique comme moyen rapide d'identifier les cas atteints de cette maladie en utilisant un code python qui suit l'architecture d'un réseau de neurones convolutif à l'aide d'algorithmes d'optimisation tels qu'AdaGrad, RMSprop, SGD ou encore Adam et cela en traitant des bases de données de radiographies thoraciques de potentielles malades et de déduire le meilleur optimiseur pour cette étude.

**Mots clés :** SARS-CoV-2, Réseau de neurones convolutif (CNN), Algorithme d'optimisation.

**ملخص:** الهدف من هذا العمل هو تصميم نظام للكشف عن فيروس كورونا ، جائحة كوفيد-19 هو جائحة لمرض معدي ناشئ يسمى مرض فيروس كورونا 2019 أو كوفيد-19 الناجم عن فيروس كورونا SARS-CoV-2 ، الذي ظهر في ووهان الصينية في 17 نوفمبر 2019 ، قبل أن ينتشر في جميع أنحاء العالم على نطاق واسع جدًا ؛ ومن هنا تأتي الحاجة إلى إنشاء نظام الكشف التلقائي كوسيلة سريعة لتحديد حالات هذا المرض باستخدام رمز بيثون يتبع بنية شبكة الخلايا العصبية التلافيفية (CNN) وبمساعدة خوارزميات التحسين مثل AdaGrad أو RMSprop أو SGD أو Adam وهذا عن طريق معالجة قواعد بيانات الأشعة السينية للصدر للمرضى المحتملين واستنتاج أفضل محسن لهذه الدراسة.

**الكلمات الأساسية:** SARS-CoV-2، شبكة العصبونات التلافيفية (CNN)، Algorithme d'optimisation.

**Summary :** The objective of this work is to design a detection system for the coronavirus, COVID-19 pandemic is a pandemic of an emerging infectious disease, called the coronavirus disease 2019 or COVID-19, caused by the coronavirus SARS-CoV-2, which appeared in Wuhan-China on 17 November 2019, before spreading worldwide on a very large scale ; hence the need to set up an automatic detection system as a rapid means of identifying cases of this disease using a python code that follows the architecture of a convolutional neuron network with the help of optimisation algorithms such as AdaGrad, RMSprop, SGD or Adam and this by processing databases of chest X-rays of potential patients and to deduce the best optimiser for this study.

**Key words :** SARS-CoV-2, Convolutional Neuron Network (CNN), Optimisation algorithm.

# Table des matières

Table des matières .....	I
Liste des abréviations.....	IV
Liste des tableaux.....	V
Liste des figures .....	VI
Introduction.....	1
Chapitre 1 Etude théorique sur la COVID-19 .....	3
1.1 Introduction .....	3
1.2 Le coronavirus (COVID-19).....	3
1.2.1 Définitions.....	3
1.2.2 Comparaison entre la COVID-19 et l'influenza .....	4
1.2.3 Transmission de la COVID-19 .....	5
1.2.4 Les symptômes.....	6
1.2.5 Le risque.....	7
1.2.6 La prévention de la transmission de la COVID-19.....	8
1.3 Etat de l'art sur les systèmes automatique de détection de la COVID-19 .....	9
1.3.1 Document 1 : Ali Narin, Ceren Kaya et Ziyne Pamuk. (2020) .....	9
1.3.2 Document 2 : Ioannis D. Apostolopoulos et Tzani A. Mpesiana .....	12
1.3.3 Document 3 : Abdolkarim Saeedi · Maryam Saeedi · Arash Maghsoudi .....	15
1.3.4 Document 4 : Morteza Heidari, Seyedehnafiseh Mirniaharikandehi1, Abolfazl Zargari Khuzani, Gopichandh Danala1, Yuchen Qiu, Bin Zheng .....	17
1.3.5 Résumé de l'état de l'art .....	18
1.4 Conclusion.....	20

Chapitre 2	Modèle proposé pour la détection de la COVID-19 .....	21
2.1	Introduction .....	21
2.2	Image x-ray .....	21
2.2.1	Définition de rayon X .....	21
2.2.2	La radiographie .....	21
2.2.3	Invention des rayons X .....	22
2.2.4	Fonctionnement des rayons X .....	23
2.2.5	L'utilisation des rayons X.....	24
2.2.6	Avantages et risques des rayons X.....	24
2.2.7	La radiographie du thorax.....	24
2.2.8	Les rayons-X et la COVID-19 .....	25
2.3	Réseau de neurones convolutif (CNN).....	26
2.3.1	Définition .....	26
2.3.2	Convolution Layer (la couche de convolution) .....	27
2.3.3	Pooling layer .....	27
	.....	29
2.4	La base de données (Dataset).....	29
2.4.1	Présentation de Dataset .....	29
2.4.2	Prétraitement et augmentation des données .....	30
2.5	Définition d'un optimiseur .....	31
2.6	Présentation des optimiseurs .....	32
2.6.1	La méthode SGD.....	32
2.6.2	La méthode RMSprop.....	33
2.6.3	La méthode Adam.....	34
2.7	Conclusion.....	36
Chapitre 3	Résultat de la détection automatique .....	37
3.1	Introduction .....	37
3.2	L'architecture du modèle utilisé.....	37
3.3	Application des optimiseurs .....	40

3.3.1	Résultat et Discussion de la méthode SGD.....	40
3.3.2	Résultats et Discussion de la méthode RMSprop .....	41
3.3.3	Résultats et Discussion de la méthode Adam .....	43
3.3.4	Tableau des résultats .....	45
3.4	Discussion .....	45
3.5	Conclusion.....	46
	Conclusion .....	47
	Bibliographie.....	48

## Liste des abréviations

Adam	Adaptive Moment Estimation
CNN	Convolution Neural Network
RMSprop	Root Mean Square Propagation
SGD	Stochastique Gradient Descent

## Liste des tableaux

Tableau 1 : Les résultats de la première étude .....	11
Tableau 2 : Les CNN de cette expérience et leurs paramètres pour l'apprentissage transfert. 13	
Tableau 3 : Résultats des CNN utilisés pour l'apprentissage par transfert. ....	13
Tableau 4 : Matrice de confusion des deux meilleurs CNN .....	14
Tableau 5 : Les TP, FP, TN et FN liés à la classe COVID-19, pour les CNN les plus .....	14
Tableau 6 : Précision, sensibilité et spécificité de MobileNet v2 sur l'ensemble de données_2 .....	14
Tableau 7 : Matrice de confusion de la classification de l'ensemble de données_2 par .....	14
Tableau 8 : Résultats du document 3 .....	16
Tableau 9 : Les résultats du document 4.....	18
Tableau 10 : Résumé de l'état d'art des systèmes automatiques de la détection de la COVID- 19.....	19
Tableau 11 : Les paramètres de l'architecture CNN utilisée. ....	39
Tableau 12 : Le résumé des résultats .....	45

# Liste des figures

Figure 1 : La comparaison entre la COVID-19 et l'influenza. ....	5
Figure 2 : Schéma fonctionnelle du document 3 .....	16
Figure 3 : Première radiographie médicale .....	23
Figure 4 : Image de rayon-X d'une personne normale .....	25
Figure 5 : Image de rayon-X d'une personne infectée par la COVID-19.....	26
Figure 6 : Exemple d'architecture d'un CNN.....	27
Figure 7 : Matrice représentatrice de l'image d'entrée .....	28
Figure 8 : Image de 2×2 pixels .....	29
Figure 9 : Commande python du prétraitement des données.....	30
Figure 10 : Commande python pour l'entraînement des données .....	31
Figure 11 : Commande python pour la validation des données.....	31
Figure 12 : L'architecture de modèle utilisé.....	38
Figure 13 : Commande python pour la méthode SGD. ....	40
Figure 14 : Matrice de confusion de SGD .....	40
Figure 15 : Les taux d'entraînement et de validation pour la méthode SGD .....	41
Figure 16 : Commande python pour la méthode RMSprop.....	41
Figure 17 : Matrice de confusion de RMSprop .....	42
Figure 18 : Les taux d'entraînement et de validation pour la méthode RMSprop.....	42
Figure 19 : Commande python pour la méthode Adam.....	43
Figure 20 : Matrice de confusion d'Adam.....	44
Figure 21 : Les taux d'entraînement et de validation pour la méthode Adam.....	44

# Introduction

Depuis 2012, les algorithmes à base d'apprentissage profond (deep learning) semblent prêts à résoudre bien des problèmes : allant de la reconnaissance faciale comme le propose DeepFace, à vaincre des joueurs de poker ou bientôt permettre la conduite de voitures autonomes ou encore la recherche de cellules cancéreuses.

Pourtant, les fondements de ces méthodes ne sont pas si récents : l'apprentissage profond a été formalisé en 2007 à partir de nouvelles architectures de réseaux de neurones dont les précurseurs sont McCulloch et Pitts en 1943 qui suivront de nombreux développements comme, les réseaux de neurones convolutifs de Yann Le Cun et Yoshua Bengio en 1998 et les réseaux de neurones profonds qui en découlent en 2012 et ouvrent la voie à de nombreux champs d'application comme la vision, le traitement du langage ou la reconnaissance de la parole.

Une épidémie de pneumonies d'allure virale d'étiologie inconnue a émergé dans la ville de Wuhan (province de Hubei, Chine) en décembre 2019. Le 9 janvier 2020, la découverte d'un nouveau coronavirus (d'abord appelé 2019-nCoV puis officiellement SARS-CoV-2, différent des virus SARS-CoV, responsable de l'épidémie de SRAS en 2003 et MERS-CoV, responsable d'une épidémie évoluant depuis 2012 au Moyen-Orient) a été annoncée officiellement par les autorités sanitaires chinoises et l'Organisation mondiale de la santé (OMS). Ce nouveau virus est l'agent responsable de cette nouvelle maladie infectieuse respiratoire appelée COVID-19.

Dans ce mémoire, nous allons concevoir un système automatique pour la détection rapide de cette maladie contagieuse. Ce système est sous forme d'un code python qui suit l'architecture d'un réseau de neurones convolutif.

Dans notre étude nous avons besoin d'une base de données qui contient des images de radiographie pour entrainer notre système à la détection des cas positive à la COVID-19.

Nous allons explorer l'optimisation en testant plusieurs optimiseurs afin de choisir celui avec la meilleure résolution qui conviendra à notre programme.

Dans le premier chapitre nous allons présenter la COVID-19 d'une vue scientifique et visiter un état de l'art sur différents travaux conçu dans le même domaine que notre étude.

Dans le deuxième chapitre nous allons définir ce que les rayon-X, la radiographie en générale et la radiographie du thorax en particulier, ensuite nous allons expliquer ce qu'un réseau de neurones convolutif, ce qu'une base de données et ce qu'un optimiseur, nous présenterons aussi différents optimiseurs.

Dans le troisième chapitre on donnera les résultats des applications avec plusieurs optimiseurs, les valeurs obtenus nous permettrons de discuter les différences et d'en déduire le meilleur optimiseur.

# Chapitre 1

## Etude théorique sur la COVID-19

### 1.1 Introduction

La pandémie du nouveau coronavirus (COVID-19) est apparue à Wuhan, en Chine, en décembre 2019 et est devenu un grave problème de santé publique dans le monde entier. Le virus qui a causé la maladie pandémique COVID-19 a été appelée coronavirus 2 du syndrome respiratoire aigu sévère, également appelée SARS-CoV-2. Les coronavirus (CoV) sont une grande famille de virus qui provoquent les maladies résultantes de rhumes telles que le syndrome respiratoire du Moyen-Orient (MERS-CoV) et le Syndrome respiratoire aigu sévère (SRAS-CoV).

La détection rapide de ce virus est très importante dont l'intérêt de créer un système automatique pour effectuer cette tâche.

Nous allons présenter la COVID-19 d'une vue scientifique, et pour concevoir un système de détection de la COVID-19, il faut connaître les différents travaux ultérieurs qui ont été réalisés. Dans ce chapitre, nous allons décrire ces travaux en précisant la performance et les résultats obtenus pour chaque recherche.

### 1.2 Le coronavirus (COVID-19)

#### a) Définitions

##### 1.2.a.1 L'épidémie et la pandémie

L'épidémie signifie la propagation rapide d'une maladie dans une population ou une région ou une zone géographique.

Une pandémie est une épidémie survenant dans le monde entier ou dans une très grande zone. Parmi les pandémies les plus connus, on retrouve la peste noire qui a sévi au Moyen Âge, grippe espagnole de 1918, virus de l'immunodéficience humain (VIH) qui a survécu depuis les années 80 et la grippe A (H1N1) de 2009.

### **1.2.a.2 Le coronavirus (COVID-19)**

En termes d'impact sur les activités humaines, la pandémie de coronavirus (COVID-19) a été sans précédent au cours des 100 dernières années. La COVID-19 est causé par le syndrome respiratoire aigu sévère coronavirus 2 (SRASCoV-2). En décembre 2019, le premier lot de cas de COVID-19 est apparu à Wuhan, en Chine, où des personnes ont présenté des symptômes graves de pneumonie (par exemple fièvre, toux, difficulté à respirer, hémoptysie). En janvier 2020, le virus s'est propagé en Asie, en Europe et aux Amériques. Le 11 mars 2020, l'Organisation mondiale de la santé (OMS) a déclaré la COVID-19 comme une pandémie car 114 pays ont dénombré la maladie. Des millions de cas on était déclaré à travers le monde. Cependant, le nombre réel d'infections peut être plus élevé, car le nombre de tests effectués dans certains pays est très faible et certaines personnes asymptomatiques peuvent ne pas être diagnostiquées.

### **1.2.a.3 L'influenza**

L'influenza (ou la grippe) est une maladie infectieuse fréquente et contagieuse. Les principaux symptômes sont de la fièvre, des courbatures, une toux sèche et une grande fatigue.

### **b) Comparaison entre la COVID-19 et l'influenza**

Beaucoup de gens veulent connaître la différence Entre la COVID-19 et grippe saisonnière. Il y a quatre points principaux à distinguer. Premièrement, le taux de reproduction du SARS-CoV-2 est 2 à 5 fois plus élevé que le virus de la grippe. Celui-là indique que les personnes infectées par la COVID-19 infecteront en moyenne 2 à 5,7 personnes, et la grippe saisonnière infectera 1 à 3 personnes. Deuxièmement, le taux de létalité (c.-à-d. les gens qui meurent de cette maladie) de COVID-19 est environ 10 à 20 fois plus élevé que la grippe saisonnière. Troisièmement, Le temps d'incubation de la COVID-19 est de 2 à 14 jours, Beaucoup plus long que la grippe saisonnière. Enfin, le taux d'hospitalisation pour la COVID-19 est environ 3 à 10

fois plus élevé que pour la grippe saisonnière, Parce que la COVID-19 provoque généralement des complications comme des pneumonies. La figure suivante résume ce qu'on vient de dire :

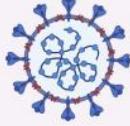
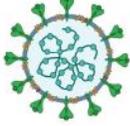
Maladie	Maladie à coronavirus 2019 (COVID-19)	Grippe saisonnière	Syndrome respiratoire aigu sévère (SRAS)	Syndrome respiratoire du Moyen-Orient (MERS)
Virus causant la maladie	 SRAS-CoV-2	 Virus de l'influenza	 SRAS-CoV	 MERS-CoV
Taux de reproduction ( $R_0$ )	2 à 5,7 <sup>a</sup>	1,3	3	0,3 à 0,8
Taux de létalité	Autour de 1 % <sup>b</sup>	0,05 à 0,1 %	9,6 à 11 %	34,4 %
Temps d'incubation	2 à 14 jours <sup>c</sup>	1 à 4 jours	2 à 7 jours	6 jours
Taux d'hospitalisation	6 à 19 % <sup>d</sup>	2 %	La plupart des cas	La plupart des cas
Taux potentiel d'infection communautaire	35 à 80 % <sup>e</sup>	10 à 20 %	10 à 60 %	4 à 13 %

Figure 1 : La comparaison entre la COVID-19 et l'influenza.

### c) Transmission de la COVID-19

Une maladie infectieuse telle que la COVID-19 se transmet lorsque six éléments d'une chaîne sont réunis. D'abord, un agent infectieux, soit le virus SRAS-CoV-2 (élément 1), colonise un environnement vivant ou inanimé, appelé « hôte » (élément 2). Les scientifiques estiment qu'il est probable que la chauve-souris ait été le premier hôte du SRAS-CoV-2 avant que ce virus ne subisse des mutations et passe à des hôtes intermédiaires. Les serpents, les pangolins et les tortues sont présentement identifiés comme des hôtes intermédiaires probables, quoique cela demeure toujours incertain. La voie de sortie empruntée par le virus (c.-à-d. le site anatomique spécifique pour quitter l'hôte animalier) (élément 3), le mode de transmissions utilisées (élément 4), et la voie d'entrée (élément 5) vers un second hôte réceptif (élément 6), soit l'être humain, demeurent toujours inconnus. La consommation humaine d'une viande animale contaminée par le virus SRAS-CoV-2 est une des hypothèses avancées pour expliquer cette transmission de l'animal vers l'humain.

La chaîne de transmission du virus SRAS-CoV-2 entre humains est toutefois mieux documentée. Les récepteurs de l'enzyme de conversion de l'angiotensine 2 (ECA2) situés au niveau des cellules épithéliales pulmonaires sont identifiés comme le site d'entrée du virus chez l'humain. Les sécrétions respiratoires sont identifiées comme la voie de sortie du virus alors que celui-ci se transmet principalement par le biais de gouttelettes respiratoires produites par une personne infectée. L'infection peut aussi être transmise par le biais d'un contact étroit et prolongé avec une personne contaminée. Lorsqu'une personne est atteinte de la COVID-19, on considère qu'elle est contagieuse environ une journée avant l'apparition des symptômes et jusqu'à 14 jours à la suite de l'apparition de ceux-ci. Il est également possible de contracter l'infection en touchant des surfaces contaminées par le virus, puis en touchant la « Zone T » de son visage, soit les yeux, le nez et la bouche. Lorsque les conditions sont favorables, le SRAS-CoV-2 peut survivre jusqu'à trois heures dans l'air, jusqu'à 24 heures sur le carton et jusqu'à deux à trois jours sur le plastique et l'acier inoxydable. À ce jour, aucune donnée scientifique ne suggère que les animaux de compagnie puissent être infectés et transmettre la COVID-19.

#### **d) Les symptômes**

Les symptômes de la COVID-19 ne sont pas spécifiques et la présentation de la maladie peut varier de l'absence de symptômes (patients asymptomatiques), à la pneumonie sévère et la mort. Dans la majorité des cas (environ 80 %), les personnes infectées par la COVID-19 présentent des symptômes légers à modérés (p. ex., toux, fièvre, fatigue) alors que 14 % d'entre eux ont des symptômes sévères (p. ex., dyspnée et hypoxémie), et 6 % présentent un tableau clinique critique (p. ex., insuffisance respiratoire, choc septique, insuffisance multi organique).

On résume les principaux symptômes, avec leur pourcentage d'apparition :

- Fièvre (87,9 %)
- Frissons (11,4 %)
- Toux (67,7 %)
- Nausée/Vomissements (5,0 %)
- Dyspnée (18,6 %)
- Hémoptysie (0,9 %)
- Myalgie (14,8 %)
- Congestion conjonctivale (0,8 %)

- Fatigue (38,1 %)
- Mal de gorge (13,9 %)
- Congestion nasale (4,8 %)
- Asymptomatique (% inconnu)
- Expectoration (33,4 %)
- Céphalée (13,6 %)
- Diarrhée (3,7 %)
- Perte de l'odorat (anosmie) et du goût (agueusie) (% inconnu)

### e) **Le risque**

Les personnes considérées les plus à risque de décéder à la suite de complications liées à la COVID-19 sont les personnes âgées généralement de plus de 65 ans, ainsi que les personnes atteintes d'une condition de santé sous-jacente ou une maladie chronique, comme l'hypertension, le diabète, les maladies cardiovasculaires, les maladies respiratoires chroniques et le cancer.

L'information sur la COVID-19 demeure limitée chez les enfants et les adolescents. Selon les données disponibles, ceux-ci représentent un faible pourcentage des cas, soit moins de 1 %. En général, la gravité et le taux de mortalité de cette maladie sont moindres dans cette tranche d'âge que chez les adultes. Tout comme les adultes, les enfants et les adolescents peuvent présenter des symptômes tels que la fièvre, une toux sèche et de la fatigue, ainsi que des douleurs abdominales et de la diarrhée dans certains cas.

Jusqu'à présent, les données disponibles sur la COVID-19 ne sont pas assez pour prouver s'il y a un risque plus grand concernant les femmes enceintes et les bébés. Cependant, selon quelques cas signalés, les caractéristiques cliniques de la COVID-19 pour les femmes enceintes sont similaire à ceux rapportés chez les patients adultes. À ce jour, rien ne prouve que le virus puisse être transmis de la mère au bébé pendant la grossesse (propagation verticale). Aucun virus n'a été détecté dans le liquide amniotique, placenta ou lait maternel des femmes enceintes infecter. De plus, il n'y a pas de déformation congénitale causée par le virus.

## **f) La prévention de la transmission de la COVID-19**

On distingue trois préventions principales :

- Les mesures de protection personnelle :

Des Mesures sont prises pour empêcher la propagation de la COVID-19, une hygiène de base est recommandée. En effet, c'est important de souvent se laver les mains avec du savon et de l'eau au moins 20 secondes et toujours couvrir sa bouche, il faut aussi frotter le nez avec le bras ou un mouchoir lorsque vous toussiez pour réduire la propagation. Actuellement, le port du masque est recommandé uniquement pour les personnes présentant des symptômes de la COVID-19, spécialement la toux, ou pour les personnes prenant soin de personnes possiblement infectées par la COVID-19.

- Les mesures de protection de l'environnement :

Pour l'environnement, il est important de nettoyer et de désinfecter fréquemment tous les objets et toutes les surfaces qui risquent d'être contaminés, comme les poignées de portes, la robinetterie, les cellulaires et les claviers et souris d'ordinateurs. De plus, il est recommandé de minimiser le partage des objets, si possible.

- Les mesures de distanciation physique :

La distanciation physique consiste à limiter le nombre de contacts entre les personnes. Il est donc nécessaire d'éviter tous les déplacements non essentiels dans la communauté et d'éviter de se rassembler, peu importe l'occasion. Lors des déplacements essentiels, il est important de maintenir une distance d'au moins deux mètres par rapport aux autres. De plus, si une personne présente des symptômes de la COVID-19 ou qu'elle revient de voyage de l'étranger (avec ou sans symptômes), celle-ci doit s'isoler et rester à la maison au moins 14 jours afin d'éviter d'infecter d'autres personnes.

## **1.3 Etat de l'art sur les systèmes automatique de détection de la COVID-19**

Le nouveau coronavirus 2019 (COVID-19), dont le point de départ est la Chine, s'est rapidement répandu parmi les personnes vivantes dans d'autres pays, et approche plusieurs personnes dans le monde entier.

Il existe un nombre limité de kits de test COVID-19 disponibles dans les hôpitaux en raison de l'augmentation des affaires au quotidien. Il est donc nécessaire de mettre en place un système de détection automatique comme moyen rapide d'identifier les cas.

### **a) Document 1 : Ali Narin, Ceren Kaya et Ziyne Pamuk. (2020)**

Dans cette étude d'Ali Narin, Ceren Kaya et Ziyne Pamuk, trois différents modèles basés sur les réseaux neuronaux convolutifs (ResNet50, InceptionV3 et InceptionResNetV2) ont été proposés pour la détection des patients infectés par une pneumonie à coronavirus en utilisant des radiographies du thorax.

Cette étude, propose une prédiction automatique de la COVID-19 en utilisant un des modèles de transfert préformés basés sur un réseau de neurones à convolution et des images de radiographies thoraciques. Pour cela ils ont utilisé les modèles préformés ResNet50, InceptionV3 et Inception-ResNetV2 pour obtenir une plus grande précision de prédictions pour les petits ensembles de données radiographiques. La nouveauté de cet article est résumée comme suit :

- i. Les modèles proposés ont une structure de bout en bout sans méthodes d'extraction et de sélection des caractéristiques.
- ii. Les radiographies pulmonaires sont le meilleur outil pour détecter la COVID-19.
- iii. Il a été démontré que les modèles préformés donnent des résultats très élevés dans le petit ensemble de données (50 COVID-19 contre 50 Normal).

Dans cette étude, des images de radiographie pulmonaire ont été utilisées pour prédire la présence de coronavirus les patients atteints de maladies (COVID-19). Des modèles préformés populaires tels que ResNet50, InceptionV3 et Au départ, les ResNetV2 ont été formés et testés sur des images de radiographies pulmonaires.

Les résumés de cette recherche sont résumés dans le point suivant :

- Précision de la formation et des valeurs de perte pour le pli 3 des modèles préformés sont données.
- La phase de formation a été menée jusqu'à la 30e époque pour éviter le suréquipement pour tous les modèles préformés.
- La précision de formation la plus élevée est obtenu avec le modèle ResNet50, les modèles InceptionV3 et Inception-ResNetV2 ont les mêmes performances.
- ResNet50 montre un processus de formation plus rapide que les autres modèles. Bien que les modèles préformés donnent des valeurs initiales très élevées, les valeurs initiales sont inférieures à 70 en raison du faible nombre de données.
- Les valeurs de perte de formation de ResNet50, InceptionV3 et Inception Les ResNetV2 sont présentées.
- Lorsqu'ils ont analysé les chiffres des pertes, ils ont constaté que les valeurs diminuent dans trois modèles préformés pendant la phase de formation. Donc le modèle ResNet 50 permet à la fois de réduire plus rapidement les valeurs de perte et de se rapprocher de zéro.

Les comparaisons de trois modèles utilisant les données de test sont comme suivant :

- ResNet50 a donné la meilleure performance avec une précision de 98%, un rappel de 96%, et une valeur de spécificité de 100%.

- Inception-ResNetV2 a donné la plus faible performance avec une précision de 87%, un rappel de 84% et une valeur de spécificité de 90%.
- Le modèle ResNet50 offre une supériorité sur les deux autres modèles au stade de la formation et de l'expérimentation.

Le tableau suivant résume ces comparaisons :

Models/Fold		Confusion matrix and Performance results (%)								
		TP	TN	FP	FN	Acc	Rec	Spe	Pre	F1
InceptionV3	Fold-1	7	10	0	3	85	70	100	100	82
	Fold-2	10	10	0	0	100	100	100	100	100
	Fold-3	10	10	0	0	100	100	100	100	100
	Fold-4	10	10	0	0	100	100	100	100	100
	Fold-5	10	10	0	0	100	100	100	100	100
	Mean					97	94	100	100	96
ResNet50	Fold-1	8	10	0	2	90	80	100	100	89
	Fold-2	10	10	0	0	100	100	100	100	100
	Fold-3	10	10	0	0	100	100	100	100	100
	Fold-4	10	10	0	0	100	100	100	100	100
	Fold-5	10	10	0	0	100	100	100	100	100
	Mean					98	96	100	100	98
Inception-ResNetV2	Fold-1	9	7	3	1	80	90	70	75	82
	Fold-2	10	9	1	0	95	100	90	91	95
	Fold-3	10	9	1	0	95	100	90	91	95
	Fold-4	7	10	0	3	95	70	100	100	82
	Fold-5	6	10	0	4	80	60	100	100	75
	Mean					87	84	90	91	86

Tableau 1 : Les résultats de la première étude

Les abréviations de la Figure 2 sont les suivantes Vrai Positif (TP), Vrai Négatif (TN), Faux Positif (FP), Faux Négatif (FN), Précision (Acc), Rappel (Rec), Spécificité (Spe), Précision (Pre), Score F1 (F1).

Donc cette étude nous propose une approche basée sur l'apprentissage par transfert profond en utilisant des images de radiographies thoraciques obtenues de patients atteints de COVID-19 et non atteints, pour prédire les patients de la COVID-19 automatiquement. Les résultats des performances montrent que le modèle préformé ResNet50 a permis la plus grande précision de 98 % parmi les trois modèles.

### **b) Document 2 : Ioannis D. Apostolopoulos et Tzani A. Mpesiana**

Dans cette étude, un ensemble d'images radiographiques de patients atteints de pneumonie bactérienne courante, de la maladie de COVID-19 et de patients en conditions normales, a été utilisé pour la détection automatique de la maladie de coronavirus. L'objectif est d'évaluer les performances des architectures de réseaux neuronaux convolutifs de pointe proposées ces dernières années pour l'imagerie médicale de classification. Plus précisément, la procédure appelée Transfert Learning a été adoptée. Avec l'apprentissage par transfert, la détection de diverses anomalies dans de petits ensembles de données d'images médicales est un objectif réalisable, qui donne souvent des résultats remarquables. Les ensembles de données utilisés dans cette expérience sont au nombre de deux. Premièrement, une collection de 1427 images radiographiques dont 224 images de la COVID-19, 700 images de pneumonie bactérienne commune et 504 images de conditions normales. Deuxièmement, un ensemble de données dont 224 images avec la maladie de COVID-19, 714 images avec la pneumonie bactérienne et virale, et 504 images de conditions normales. Les données ont été recueillies à partir des images radiographiques disponibles sur les dépôts médicaux publics. Les résultats suggèrent que l'apprentissage profond avec l'imagerie radiographique peut extraire des bios marqueurs significatifs liés à la maladie de COVID-19, alors que la meilleure précision, sensibilité et spécificité obtenues sont respectivement de 96,78 %, 98,66 % et 96,46 %.

Les résultats sont donnés dans les figures suivantes :

Network	Parameter	Description
<b>VGG19</b>	<b>Layer Cutoff</b>	<b>18</b>
	<b>Neuronal Network</b>	<b>1024 nodes</b>
<b>MobileNet2</b>	<b>Layer Cutoff</b>	<b>10</b>
	<b>Neuronal Network</b>	<b>1000 nodes, 750 nodes</b>
<b>Inception</b>	<b>Layer Cutoff</b>	<b>249</b>
	<b>Neuronal Network</b>	<b>1000</b>
<b>Xception</b>	<b>Layer Cutoff</b>	<b>120</b>
	<b>Neuronal Network</b>	<b>1000 nodes, 750 nodes</b>
<b>Inception ResNet v2</b>	<b>Layer Cutoff</b>	<b>730</b>
	<b>Neuronal Network</b>	<b>No</b>

Tableau 2 : Les CNN de cette expérience et leurs paramètres pour l'apprentissage

Network	Accuracy 2-class (%)	Accuracy 3-class (%)	Sensitivity (%)	Specificity (%)
<b>VGG19</b>	<b>98.75</b>	<b>93.48</b>	<b>92.85</b>	<b>98.95</b>
<b>MobileNet2</b>	<b>97.40</b>	<b>92.85</b>	<b>99.10</b>	<b>97.09</b>
<b>Inception</b>	<b>86.13</b>	<b>92.85</b>	<b>12.94</b>	<b>99.70*</b>
<b>Xception</b>	<b>85.57</b>	<b>92.85</b>	<b>0.08</b>	<b>99.99</b>
<b>Inception ResNet v2</b>	<b>84.38</b>	<b>92.85</b>	<b>0.01</b>	<b>99.83</b>

Tableau 3 : Résultats des CNN utilisés pour l'apprentissage par transfert.

Model	Predicted labels	Actual labels		
		Actual COVID-19	Actual pneumonia	Actual normal
MobilNet v2	Predicted COVID-19	222	8	27
	Predicted pneumonia	2	495	27
	Predicted normal	0	1	646
VGG19	Predicted COVID-19	222	8	7
	Predicted pneumonia	3	460	26
	Predicted normal	13	36	667

Tableau 4 : Matrice de confusion des deux meilleurs CNN

CNN	TP	FP	TN	FN
VGG19	208	15	1189	16
MobilNet v2	222	35	1169	2

Tableau 5 : Les TP, FP, TN et FN liés à la classe COVID-19, pour les CNN les plus Performantes.

Network	Accuracy 2-class (%)	Accuracy 3-class (%)	Sensitivity (%)	Specificity (%)
MobileNet v2	96.78	94.72	98.66	96.46

Tableau 6 : Précision, sensibilité et spécificité de MobileNet v2 sur l'ensemble de données\_2

Model	Predicted labels	Actual labels		
		Actual COVID-19	Actual pneumonia	Actual normal
MobilNet v2	Predicted COVID-19	221	19	24
	Predicted pneumonia	2	472	17
	Predicted normal	1	13	673

Tableau 7 : Matrice de confusion de la classification de l'ensemble de données\_2 par

Les résultats de chaque CNN pour l'ensemble de données 1 sont illustrés dans les tableaux 2 et 3. Dans le tableau 3, la précision et la sensibilité sont présentés. Les résultats suggèrent que le

VGG19 et le MobileNetv2 peuvent obtenir une meilleure précision de classification par rapport aux autres CNN.

Les résultats du tableau 3, peuvent être simplifiés pour représenter la sensibilité et la spécificité du test concernant la COVID-19. Cette procédure a été expliquée dans la section "Métriques", les résultats sont illustrés dans le tableau 4.

L'ensemble de données 2 comprend les cas de pneumonie virale, qui sont ajoutés aux cas bactériens, et classés avec la même étiquette, c'est-à-dire la pneumonie commune (classe unique). La mise en place de MobileNet v2 de la première expérience a été retenue pour effectuer la tâche de classification de l'ensemble de données 2. Les résultats sont illustrés dans les tableaux des figures 5 et 6.

### **c) Document 3 : Abdolkarim Saeedi · Maryam Saeedi · Arash Maghsoudi**

Dans ce travail, Les trois auteurs introduisent un service web de diagnostic assisté par ordinateur (CAD) pour détecter la COVID-19 en ligne. L'une des plus grandes bases de données publiques de scanners thoraciques, contenant 746 participants, a été utilisée dans cette expérience. Un certain nombre d'architectures de réseaux neuronaux profonds bien connues comprenant ResNet, Inception et MobileNet ont été inspectés afin de trouver le modèle le plus efficace pour le système hybride.

Dans ce document, ils nous proposent un pipeline rapide utilisant une combinaison des algorithmes d'apprentissage profond et d'apprentissage automatique. DenseNet121 présente des résultats remarquables basés sur le classificateur Nu-SVM. L'hybride introduit est également comparé à d'autres modèles des méthodes d'apprentissage pour affirmer une des preuves empiriques de la fiabilité de la combinaison proposée. Les modèles sont également hébergés sur un service web en ligne, pour faire une détection en direct basée sur les images du scanner thoracique, en fournissant le code source et une explication, ils insistent en outre pour fournir suffisamment d'informations pour que la communauté puisse continuer avec la pandémie. Dans ce travail, les algorithmes divergents d'apprentissage profond ont été inspectés avec le Nu-SVM. La Figure suivante représente le schéma fonctionnel de l'ensemble du processus.

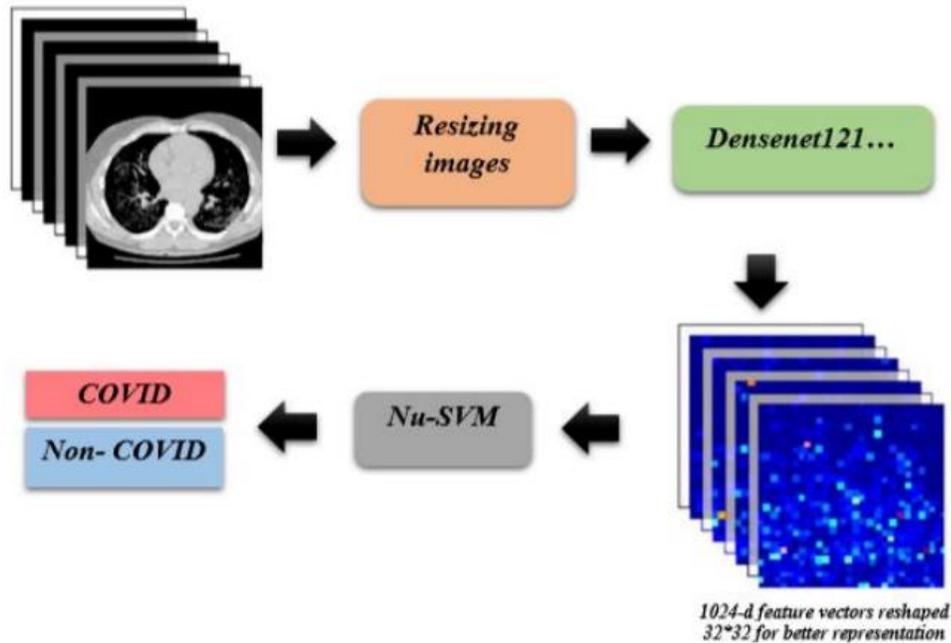


Figure 2 : Schéma fonctionnelle du document 3

Tout le traitement est effectué en utilisant python et un nombre de cadres, à savoir Tensorflow, Keras, Scikit-learn. Des structures profondes communes composé de DenseNet121, ResNet50 V1&V2, InceptionV3 et MobileNet V1&V2 ont été utilisés après la phase de redimensionnement des images pour tirer des caractéristiques à partir des images.

Les résultats de cette étude sont présentés dans le tableau suivant :

	Accuracy	Recall	Precision	F1-Score	AUC	Output Size
<b>DenseNet21</b>	<b>90.61</b> ( $\pm 5.4$ )	<b>90.80</b> ( $\pm 5.3$ )	<b>89.76</b> ( $\pm 3.7$ )	<b>90.13</b> ( $\pm 5.4$ )	<b>95.05</b> ( $\pm 2.9$ )	<b>1024</b>
<b>InceptionV3</b>	<b>81.63</b> ( $\pm 3.7$ )	<b>80.08</b> ( $\pm 5.3$ )	<b>80.07</b> ( $\pm 6.3$ )	<b>80.49</b> ( $\pm 3.4$ )	<b>87.78</b> ( $\pm 3.1$ )	<b>2048</b>
<b>ResNet50V2</b>	<b>84.03</b> ( $\pm 4.6$ )	<b>82.23</b> ( $\pm 7.5$ )	<b>83.83</b> ( $\pm 5.7$ )	<b>82.76</b> ( $\pm 4.9$ )	<b>84.49</b> ( $\pm 3.7$ )	<b>2048</b>
<b>ResNet50V1</b>	<b>73.72</b> ( $\pm 3.9$ )	<b>73.94</b> ( $\pm 6.5$ )	<b>71.85</b> ( $\pm 3.6$ )	<b>72.46</b> ( $\pm 3.6$ )	<b>79.18</b> ( $\pm 5.3$ )	<b>2048</b>
<b>MobileNetV1</b>	<b>89.14</b> ( $\pm 2.6$ )	<b>88.53</b> ( $\pm 4.2$ )	<b>88.64</b> ( $\pm 5.1$ )	<b>88.43</b> ( $\pm 2.7$ )	<b>94.15</b> ( $\pm 1.9$ )	<b>1024</b>
<b>MobileNetV2</b>	<b>85.52</b> ( $\pm 3.7$ )	<b>87.66</b> ( $\pm 4.4$ )	<b>82.84</b> ( $\pm 5.5$ )	<b>85.03</b> ( $\pm 3.6$ )	<b>92.78</b> ( $\pm 2.7$ )	<b>1024</b>

Tableau 8 : Résultats du document 3

La méthodologie proposée a permis d'obtenir un taux de rappel de 90,80 %, une précision de 89,76 % et une exactitude de 90,61 %.

**d) Document 4 : Morteza Heidari, Seyedehnafiseh Mirniaharikandehi1, Abolfazl Zargari Khuzani, Gopichandh Danala1, Yuchen Qiu, Bin Zheng**

Les auteurs de cette étude ont développé et tester un nouveau système de diagnostic assisté par ordinateur (CAD). Il comprend plusieurs prétraitements d'images des algorithmes permettant de supprimer les diaphragmes, de normaliser le rapport contraste/bruit de l'image et de générer trois images en entrées puis des liens vers un réseau neuronal convolutif basé sur l'apprentissage par transfert (un réseau CNN basé sur le VGG16) pour classer les radiographies pulmonaires en trois classes de pneumonie infectée par la COVID-19.

Un ensemble de données de 8 474 radiographies pulmonaires est utilisé, dont 415 confirmées comme étant infectées par la COVID-19, 5 179 cas de pneumonie communautaire et 2 880 cas de non-pneumonie. L'ensemble de données est divisé en deux sous-ensembles avec 90 % et 10% d'images dans chaque sous-ensemble pour former et tester le programme de CAD. Les résultats des tests atteignent 94,0% de précision globale dans la classification de trois classes et 98,6% dans la détection des cas d'infection par la COVID-19. Ainsi, l'étude démontre la possibilité de développer un système de CAD pour les radiographies du thorax et fournissant aux radiologues des outils utiles d'aide à la décision dans la détection et le diagnostic de la pneumonie infectée par la COVID-19.

Dans cette étude, ils utilisent un réseau neuronal profond VGG16 et l'appliquent à un nouveau schéma CAD des radiographies pulmonaires pour détecter et classer les images en 3 classes, à savoir les images infectées par la COVID-19, les autres pneumonies virales d'origine communautaire et les cas normaux (autres que la pneumonie). Ils appliquent une approche d'apprentissage par transfert, Plus précisément le VGG16 original qui comprend plus de 138 millions de paramètres. Ces paramètres ont été formés et déterminés à l'aide d'une grande base de données ImageNet, qui contient plus de 14 millions d'images.

Les résultats de l'étude démontrent que cette approche d'apprentissage par transfert peut donner de meilleures performances avec la précision globale de 93,9% (796/848) dans la classification de trois classes et de 98,6% (836/848) dans la classification des cas avec et sans la maladie de la COVID-19, ainsi que la grande robustesse avec un score kappa de Cohen de 0,88 comme indique le tableau suivant :

	Precision	Recall	F1-Score	Support Cases
Normal	0.94	0.90	0.92	288
Other Pneumonia	0.95	0.95	0.95	518
COVID-19	0.78	1.00	0.88	42
Accuracy	---	---	0.94	848
Macro avg	0.89	0.95	0.92	848
Weighted avg	0.94	0.94	0.94	848

Tableau 9 : Les résultats du document 4

Dans l'étape suivante, à partir de la matrice de confusion, ils calculent et déterminent la précision, le taux de rappel, F1 et la précision des prévisions du nouveau modèle CNN basé sur le VGG16, comme indiqué dans le tableau 10. Parmi les 848, 796 cas sont correctement détectés et classés en trois classes. Ainsi, la précision globale est de 93,9% (796 / 848) avec un intervalle de confiance de 95% de [0,92, 0,96]. En outre, le coefficient kappa de Cohen pour la méthode proposée est de 0,88 ce qui confirme la fiabilité de l'approche proposée pour former ce nouveau modèle d'apprentissage profond pour effectuer cette tâche de classification. Après ils ont classé les images de pneumonie normale et de pneumonie communautaire dans la classe négative et les cas de pneumonie infectée par la COVID-19 dans la classe positive. En combinant les données de la matrice de confusion et le schéma CAD, ils en concluent une sensibilité de 100% (42/42) et une spécificité de 98,5% (794/806). La précision globale est de 98,6 % (836/848).

### e) Résumé de l'état de l'art

On résume les résultats des travaux précédents dans un tableau :

<b>Auteur et année</b>	<b>Objective</b>	<b>Méthode utilise</b>	<b>Résultats</b>
Ali Narin, Ceren Kaya et Ziyet Pamuk. (2020)	Testé trois modèles de CNN pour la détection automatique de la COVID-19	Trois différents modèles basés sur les réseaux neuronaux convolutifs (ResNet50, InceptionV3 et InceptionResNetV2)	ResNet50 a donné la meilleure performance avec une précision de 98%, un rappel de 96%, et une valeur de spécificité de 100%
Ioannis D. Apostolopoulos et Tzani A. Mpesiana (2020)	Evaluer les performances des architectures de réseaux neuronaux convolutifs	Transfert learning	La meilleure précision, sensibilité et spécificité obtenues sont respectivement de 96,78 %, 98,66 % et 96,46 %
Abdolkarim Saeedi, Maryam Saeedi et Arash Maghsoudi (2020)	Construire un service web de diagnostic assisté par ordinateur (CAD) pour détecter la COVID-19 en ligne	Plusieurs architectures de CNN (DenseNet121 InceptionV3 ResNet50V2 ResNet50V1 MobileNetV1 MobileNetV2)	La meilleure architecture DenseNet121 qui a donnée taux de rappel de 90,80 %, une précision de 89,76 % et une exactitude de 90,61 %.
Mortaisa Heidari, Seyedehnafiseh Mirniaharikandehei, Abolfazl Zargari Khuzani, Gopichandh Danala, Yuchen Qiu et Bin Zheng	Développer et tester un nouveau système de diagnostic assisté par ordinateur (CAD)	Un réseau CNN basé sur le VGG16	Précision globale de 93,9%

Tableau 10 : Résumé de l'état d'art des systèmes automatiques de la détection de la COVID-

## **1.4 Conclusion**

Dans ce chapitre nous avons différencié l'épidémie de la pandémie, Nous avons expliqué la différence entre l'influenza et la Coronavirus, Nous avons présenté la COVID-19 d'un point de vue scientifique et Nous avons donné un aperçu sur l'état de l'art des systèmes automatiques utilisé pour la détection de cette maladie qui sont basé sur les réseaux de neurones convolutifs (CNN).

# Chapitre 2

## Modèle proposé pour la détection de la COVID-19

### 2.1 Introduction

Dans cette étude nous allons utiliser l'imagerie aux rayons-X (x-ray), des radiographies comme base de données, nous aurons aussi besoin d'un réseau de neurones convolutif pour l'entraînement de notre système automatique.

Dans ce chapitre nous allons présenter les images X-ray avec des exemples de cas positifs à la COVID-19 et les comparer aux cas normaux. Nous allons aussi présenter le réseau de neurones convolutif et ces deux couches principales, Nous allons expliquer ce qu'une base de données et définir ce qu'un optimiseur, pour ensuite présenter différents optimiseurs.

### 2.2 Image x-ray

#### a) Définition de rayon X

Découvert en 1895 par le physicien Wilhelm Röntgen. Les rayons X sont une forme de rayonnement électromagnétique à haute fréquence constitué de photons dont l'énergie varie de 5 picomètres à 10 nanomètres et qui pénètrent la matière condensée. Utilisé dans la cristallographie (Science des formes, des structures et des propriétés des cristaux) et l'imagerie médicale.

#### b) La radiographie

La radiographie est une technique d'imagerie de transmission, par rayons X dans le cadre de la radiographie X, et les technologues en radiations médicales sont des professionnels qui créent ces images aux rayons X. Cette technologie est la plus utilisée dans le domaine de la santé. Les images aux rayons X (images statiques similaires aux photographies), les images de

radioscopie (images en mouvement similaires à un film) et les images tomographiques (images en coupe) ne sont que quelques façons de visualiser l'organisme à l'aide de rayons X.

Les techniciens en radiation médicale doivent avoir une bonne compréhension de la biologie et des sciences physiques. D'un point de vue biologique, ils s'intéressent principalement à la physiologie et à l'anatomie humaine, avec une attention particulière à l'orthopédie liée aux os. En étudiant la physique et la chimie, ils ont également appris comment les rayons X sont produits et comment leur faisceau change lorsqu'ils interagissent avec le corps humain pour produire des images.

### **c) Invention des rayons X**

C'est le physicien Allemand Wilhelm Conrad Roentgen qui a découvert les rayons X le 8 novembre 1895, alors qu'il travaillait avec des tubes à rayons cathodiques (genre de cylindres sous vide et scellés dotés d'un canon à électrons à une extrémité et d'un écran conçu pour capter des images à l'autre). Roentgen avait alors remarqué que l'écran qu'il utilisait (une plaque photographique composée d'un produit chimique appelé « platinocyanure de baryum ») devenait fluorescent (émettait de la lumière), et le restait même après qu'il ait bloqué les rayons cathodiques. Il a baptisé ce nouveau type de radiation électromagnétique le « rayonnement X », parce qu'il ne savait pas vraiment de quoi il s'agissait et qu'en mathématique, le « x » représente une valeur inconnue. On a plus tard raccourci le terme en utilisant l'expression « rayons X ». Ceux que Roentgen a découverts étaient très pénétrants s'ils passaient au travers des tissus mous, ils ne traversaient pas aussi facilement les tissus durs (comme les os) et le métal (comme la bague de sa femme).

Il a également découvert que ces rayons peuvent provoquer l'apparition des images sur de la pellicule photographique. En reconnaissance de ses travaux novateurs, Roentgen a reçu de nombreuses distinctions, dont le premier prix Nobel de physique en 1901. L'unité de mesure qui a longtemps été utilisée pour démontrer le niveau de l'exposition aux rayons X (R, pour röntgen) a été nommée en son honneur. Aujourd'hui, nous utilisons des Coulombs par kilogramme (C/kg). La figure suivante montre la première radiographie médicale :



Figure 3 : Première radiographie médicale

#### d) **Fonctionnement des rayon X**

Les appareils de radiographie sont essentiellement des caméras spécialisées. Dans leur cas, au lieu de la lumière visible, ce sont des rayons X qui exposent une pellicule de façon à créer des images. Les tissus osseux, qui contiennent beaucoup de calcium, absorbent ces rayons, alors que les tissus mous, comme la peau et le gras, les laissent passer. Sur une radiographie, les zones sombres correspondent donc aux tissus moins compacts, alors que les zones plus claires représentent les endroits où les tissus sont plus denses ou épais.

### **e) L'utilisation des rayons X**

Les rayons X ne sont pas seulement utilisés pour vérifier l'état des os. Le dentiste par exemple les utilise pour rechercher des caries dans les dents. Des technologues en emploient également des versions à faible énergie pour réaliser des mammographies, qui aident les médecins à dépister des maladies du sein. Le personnel de sécurité de l'aéroport l'utilise également pour détecter les objets dangereux dans les bagages des passagers. Ces rayons sont également utilisés pour les tests non destructifs (tests destinés à déterminer la faiblesse d'un objet sans le détruire). Par exemple, les rayons X peuvent même révéler des défauts cachés, tels que de minuscules fissures ou de mauvaises soudures sur les tuyaux.

### **f) Avantages et risques des rayons X**

L'imagerie par rayons X est très utile pour examiner les os et d'autres tissus durs, tels que les calculs biliaires (parfois des cristaux formés dans la vésicule biliaire), les calculs rénaux (également des cristaux formés cette fois-ci dans les reins), les tumeurs cancéreuses et les objets avalés accidentellement (Clés, pièces de monnaie, épingles, etc.). Cependant, comme les rayons X sont une forme de rayonnement ionisant, ils augmentent le risque de cancer. Cependant, la dose reçue lors de l'examen radiographique chez le dentiste ou à l'hôpital est très faible.

### **g) La radiographie du thorax**

La radiographie du thorax est une radiographie pulmonaire qui montre le cœur, les poumons, les vaisseaux sanguins et les os. La radiographie consiste à projeter des rayons X sur le corps humain pour créer une image sur un tableau de capture numérique. Différentes parties du corps humain absorbent différentes quantités de rayonnement. Les parties qui absorbent le plus de rayonnement (comme les os) apparaissent blanches sur l'écran. Les parties qui absorbent moins de rayonnement (comme le cœur) sont représentées dans différentes nuances de gris. Les zones contenant de l'air (comme les poumons) sont indiquées en noir. Les images radiographiques peuvent montrer la présence d'une hypertrophie du cœur et d'un œdème pulmonaire (liquide dans les poumons) - symptômes associés à une insuffisance cardiaque.

Des images plus détaillées sont généralement nécessaires pour confirmer les anomalies observées sur la radiographie pulmonaire.

### **h) Les rayons-X et la COVID-19**

Les appareils à rayons X sont utilisés pour scanner le corps affecté, comme les fractures, les dislocations osseuses, les infections pulmonaires, les pneumonies et les tumeurs. L'utilisation des rayons X est une méthode rapide, facile, moins coûteuse et moins nocive pour la détection rapide de la COVID-19, c'est pour ça qu'on va les utiliser pour notre système automatique. Les figures suivantes montrent respectivement des images (X-ray) de personnes normales et de personnes infectées par la COVID-19 :



Figure 4 : Image de rayon-X d'une personne normale

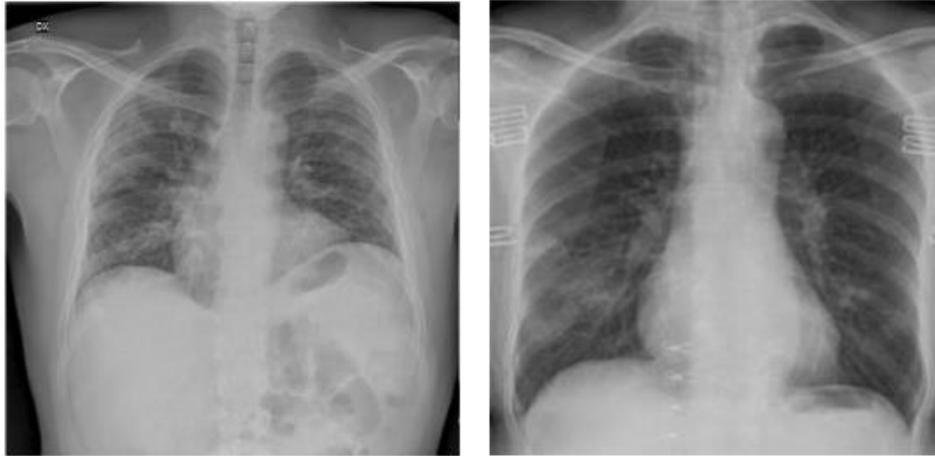


Figure 5 : Image de rayon-X d'une personne infectée par la COVID-19

## 2.3 Réseau de neurones convolutif (CNN)

### a) Définition

Le CNN est un réseau de neurones profond qui contient plusieurs couches cachées et qui est utilisé pour la reconnaissance des images et les problèmes de classification des images. Ce réseau est un type de modèle d'apprentissage profond presque universellement utilisé dans les applications de vision par ordinateur.

Pour le côté optimisation dans ce réseau on a plusieurs optimiseur a utilisé tel que le SGD, Adam et RMSprop...etc. tous ces optimiseurs sont basés sur le SGD, le but de cette méthode est de trouver le minimum d'une fonction différentiable (un point où la dérivée est nul), pour le réseau de neurones convolutif, cela signifie qu'il faut trouver analytiquement la combinaison des valeurs de poids qui donnent la plus petite fonction de pertes possible.

La fonction de pertes et aussi nécessaire pour le réseau de neurones convolutif, le but de cette fonction est de trouver la quantité qu'un modèle doit chercher à minimiser durant le processus d'entraînement.

## b) Convolution Layer (la couche de convolution)

La couche de convolution est la pierre angulaire du réseau convolutif qui effectue la majeure partie du travail de calcul, vue d'ensemble et intuition sans cervelle. Regardons d'abord ce que la couche CONV calcule sans l'analogie cerveau / neurone. Les paramètres de la couche CONV consistent en un ensemble de filtres d'apprentissages. Comprend des filtres qui convertissent les images. Nous appellerons ces filtres des filtres de convolution. Chaque filtre a un petit espace (largeur et hauteur) mais s'étend sur toute la profondeur du volume d'entrée. La couche de convolution génère de nouvelles images appelées « cartes de caractéristiques ». La carte met en évidence les caractéristiques uniques de l'image originale. Cette Couche fonctionne complètement différemment des autres couches du réseau neuronal elle n'utilise pas le poids du lien et la somme pondérée, et le processus d'introduction de l'image à travers les filtres de convolution produit une carte de caractéristiques. la figure suivante explique le processus de la couche de convolution :

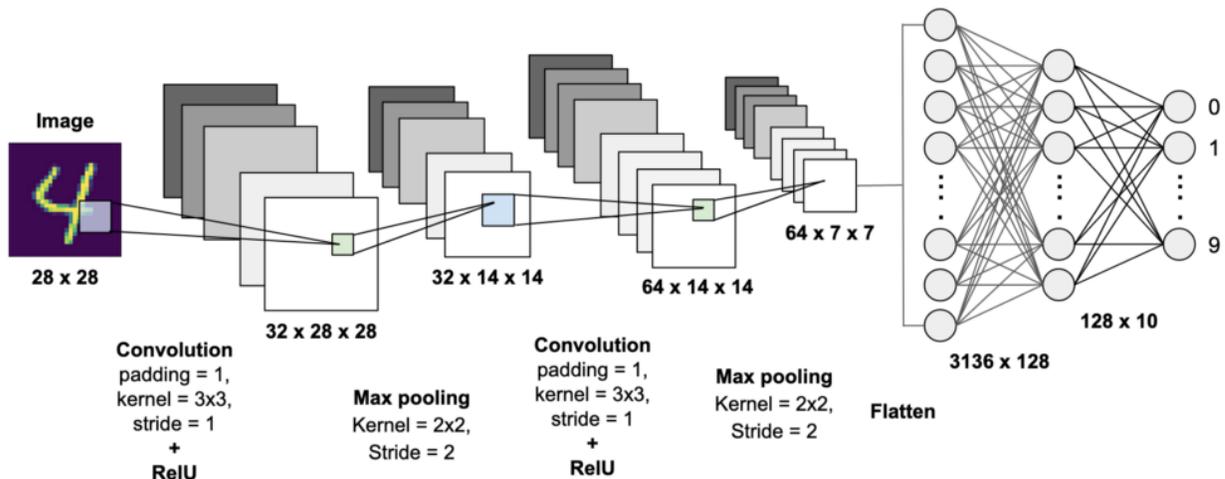


Figure 6 : Exemple d'architecture d'un CNN

## c) Pooling layer

C'est une couche de regroupement qui réduit la taille de l'image car elle combine des images de pixels adjacentes d'une zone spécifique de l'image en une seule valeur représentative. Cette technique est typiquement adoptée par de nombreux autres systèmes de traitement d'images.

Afin d'opérer dans la couche de pooling, nous devons déterminer Comment choisir les pixels groupés de l'image et comment définir le représentant. Sélectionnez généralement les pixels adjacents dans une matrice carrée, puis le nombre de pixels fusionnés varie en fonction du problème. La valeur représentative est généralement définie comme la valeur moyenne ou maximale des pixels sélectionnés. Le fonctionnement de la couche de pooling est très simple.

Prenons un exemple. Une image d'entrée de  $4 \times 4$  pixels, représentée par la matrice illustrée dans la figure ci-dessous :

1	1	1	3
4	6	4	8
30	0	1	5
0	2	2	4

Figure 7 : Matrice représentatrice de l'image d'entrée

Nous fusionnons les pixels de l'image d'entrée dans une matrice  $2 \times 2$  **Erreur ! Cela devrait être un chiffre.** sans éléments qui se chevauchent. Une fois que l'image d'entrée a traversé la couche de regroupement, Il est réduit à une image de  $2 \times 2$  pixels. La figure suivante montre la fusion des résultats utilisés une moyenne pooling (mean pooling) et un max pooling.

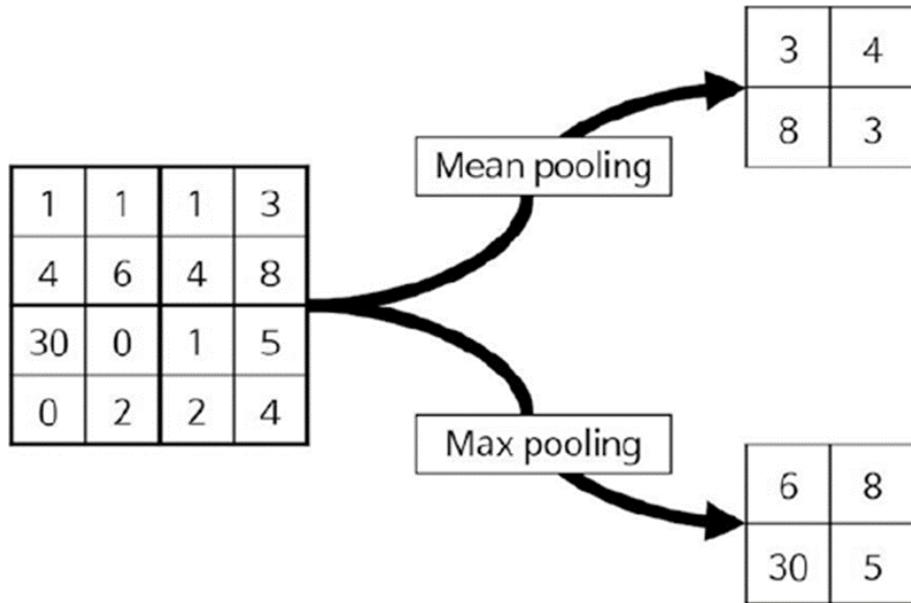


Figure 8 : Image de 2×2 pixels

## 2.4 La base de données (Dataset)

### a) Présentation de Dataset

Dans notre étude on utilise un ensemble de données publiques de radiographies pulmonaires et de tomodensitogrammes de patients positifs ou suspectés d'être atteints de la COVID-19 ou d'autres pneumonies virales et bactériennes (MERS, SARS et ARDS). Les données seront collectées auprès de sources publiques ainsi que par collecte indirecte auprès des hôpitaux et des médecins.

L'ensemble de données comprend 284 images radiographiques 2D en vue postéro-antérieure (P.A.) de la poitrine. Parmi lesquelles nous avons 112 images où la maladie de COVID-19 est confirmée, 112 cas normaux (non pneumonie) pour la partie d'entraînement ; et 30 images de cas confirmés de la COVID-19 avec 30 images aussi pour les cas normaux pour la partie de validation.

## b) Prétraitement et augmentation des données

Afin de tirer la meilleure résolution de nos quelques exemples de formation, nous allons les "augmenter" par un certain nombre de transformations aléatoires, de sorte que notre modèle ne voit jamais deux fois exactement la même image. Cela permet d'éviter le sur-ajustement et aide le modèle à mieux généraliser. Dans Keras, cela peut être fait via la classe `keras.preprocessing.image.ImageDataGenerator`.

Cette classe nous permet de configurer les transformations aléatoires et les opérations de normalisation à effectuer sur nos données d'image pendant la formation d'instancier des générateurs de Batch d'images augmentées (et leurs étiquettes).

Dans notre modèle nous utilisons quelques-unes des options disponibles comme indique la figure suivante :

```
#Moulding train images
train_datagen = image.ImageDataGenerator(rescale = 1./255,
                                         shear_range = 0.2,
                                         zoom_range = 0.2,
                                         horizontal_flip = True)

test_dataset = image.ImageDataGenerator(rescale=1./255)
```

Figure 9 : Commande python du prétraitement des données

On commence par (`rescale`) le rééchelonnement, qui est une valeur par laquelle nous multiplierons les données avant tout autre traitement. Nos images originales sont constituées de coefficients RGB dans la plage 0-255, mais ces valeurs seraient trop élevées pour que nos modèles puissent les traiter (étant donné un taux d'apprentissage typique), c'est pourquoi nous ciblons plutôt des valeurs entre 0 et 1 en les mettant à l'échelle avec un facteur de  $1/255$ .

`Shear_range` est pour l'application aléatoire de transformations de cisaillements, `zoom_range` permet de zoomer aléatoirement à l'intérieur des images et `horizontal_flip` permet de retourner aléatoirement la moitié des images à l'horizontale.

Ensuite on a une partie d'entraînement de dataset où on va redimensionner les images et diviser l'ensemble de données par des batch comme indique la figure suivante :

```
train_generator = train_datagen.flow_from_directory(  
    'CovidDataset/Train',  
    target_size = (224,224),  
    batch_size = 32,  
    class_mode = 'binary')
```

Figure 10 : Commande python pour l'entraînement des données

Target\_size : Taille de nos images d'entrée, chaque image sera redimensionnée à cette taille.

Batch\_size : Nombre d'images à produire par batch à partir du générateur.

Class\_mod : le mode de la classe qui est le mode binaire car il y'a deux classes.

Ensuite on a une partie de validation avec les mêmes procédures de l'entraînement comme indique la figure suivante :

```
validation_generator = test_dataset.flow_from_directory(  
    'CovidDataset/Val',  
    target_size = (224,224),  
    batch_size = 32,  
    class_mode = 'binary')
```

Figure 11 : Commande python pour la validation des données

## 2.5 Définition d'un optimiseur

Au cours du processus d'entraînement, nous ajustons et modifions les paramètres (poids) de notre modèle pour essayer de minimiser cette fonction de perte (une façon mathématique de mesurer l'erreur de nos prévisions), et rendre nos prédictions aussi correctes et optimisées que possible.

C'est là qu'interviennent les optimiseurs. Ils relient la fonction de perte et les paramètres du modèle en actualisant le modèle en fonction des résultats de la fonction de perte. En termes plus simples, les optimiseurs façonnent et moulent votre modèle dans sa forme la plus précise possible en utilisant les poids. La fonction de perte est le guide du terrain, indiquant à l'optimiseur quand il se déplace dans la bonne ou la mauvaise direction.

Pour un modèle mental utile, vous pouvez penser à un randonneur qui essaie de descendre une montagne les yeux bandés. Il est impossible de savoir dans quelle direction aller, mais il y a une chose qu'elle peut savoir : si elle descend (elle progresse) ou si elle monte (elle perd du terrain). Finalement, si elle continue à prendre des mesures qui la mènent vers le bas, elle finira par atteindre la base.

De même, il est impossible de savoir dès le départ quels devraient être les poids de votre modèle. Mais avec quelques essais et erreurs basés sur la fonction de perte (que le randonneur descende ou non), vous pouvez finir par y arriver.

## **2.6 Présentation des optimiseurs**

### **a) La méthode SGD**

La méthode du gradient stochastique (SGD) et ses variantes sont probablement les algorithmes d'optimisation les plus couramment utilisés pour l'apprentissage automatique général, et en particulier pour l'apprentissage profond. Il est possible d'obtenir une évaluation objective du gradient en prenant un gradient moyen sur un mini-lot d'échantillons prélevés dans la distribution génératrice de données. L'algorithme SGD montre comment suivre cette estimation de chute de gradient. Le paramètre clé de l'algorithme SGD est la vitesse d'apprentissage. Le SGD utilise une vitesse d'apprentissage constante, mais dans la pratique, il est nécessaire de réduire progressivement la vitesse d'apprentissage au fil du temps. À titre de comparaison, le véritable gradient de la fonction de coût total devient petit, puis 0 lorsque nous approchons et atteignons le minimum à l'aide d'un gradient par lots.

On résume l'algorithme du SGD :

**Algorithm:** Stochastic gradient descent (SGD)

Require: Learning rate schedule  $\epsilon_1, \epsilon_2, \dots$

Require : Initial parameter  $\theta$

$k \leftarrow 1$

**While** stopping criterion not met **do**

Sample a minibatch of  $m$  examples from the training set  $\{x^{(1)} \dots x^{(m)}\}$  with corresponding targets  $y^{(i)}$ .

Compute gradient estimate:  $\hat{g} \leftarrow \frac{1}{m} \nabla_{\theta} \sum_i L(f(x^i; \theta), y^i)$

Apply update:  $\theta \leftarrow \theta - \epsilon_k \hat{g}$

$k \leftarrow k + 1$

**End while**

## b) La méthode RMSprop

RMSprop est une version spéciale d'AdaGrad développée par le professeur Geoffrey Hinton dans sa classe de filets neuronaux. Au lieu de laisser tous les gradients s'accumuler pour le Momentum, il n'accumule les gradients que dans une fenêtre fixe. RMSprop est similaire à Adaprop, qui est un autre optimiseur qui cherche à résoudre certains des problèmes qu'AdaGrad laisse ouverts.

RMSprop tente également d'amortir les oscillations, mais d'une manière différente de l'impulsion. RMSprop évite également de devoir ajuster le rythme d'apprentissage, et le fait automatiquement. De plus, RMSprop choisit un taux d'apprentissage différent pour chaque paramètre.

Dans RMSprop, chaque mise à jour est effectuée selon des équations précises. Cette mise à jour est effectuée séparément pour chaque paramètre.

**Algorithm:** The RMSprop algorithm

**Require:** Global learning rate  $\epsilon$ , decay rate  $\rho$

**Require:** Initial parameter  $\theta$

**Require:** Small constant  $\delta$ , perhaps  $10^{-6}$ , used to stabilize division by small numbers

Initialise gradient accumulation variable  $\mathbf{r} = \mathbf{0}$

While stopping criterion not met **do**

Sample a minibatch of  $m$  examples from the training set  $\{x^{(1)}, \dots, x^{(m)}\}$  with corresponding targets  $y^{(i)}$ .

Compute gradient:  $g \leftarrow \frac{1}{m} \nabla_{\theta} \sum_i L(f(x^i; \theta), y^i)$

Accumulate squared gradient:  $r \leftarrow \rho r + (1 - \rho) g \odot g$

Compute update:  $\Delta\theta \leftarrow -\frac{\epsilon}{\delta + \sqrt{r}} \odot g$  ( $\frac{\epsilon}{\delta + \sqrt{r}}$  applied element-wise)

Apply update:  $\theta \leftarrow \theta + \Delta\theta$

**End While**

### c) La méthode Adam

Adam est l'abréviation de "Adaptive Moment Estimation", dans le contexte des algorithmes précédents, cela est mieux vu comme une variante de la combinaison de RMSprop et de Momentum avec quelques différences importantes, et c'est une autre façon d'utiliser les gradients passés pour calculer les gradients actuels. Adam utilise également le concept de moment adaptatif en ajoutant des fractions de gradients précédents au gradient actuel. Cet optimiseur est devenu assez répandu, et est pratiquement accepté pour être utilisé dans les réseaux neuronaux d'entraînement.

Il est facile de se perdre dans la complexité de certains de ces nouveaux optimiseurs. Il suffit de se rappeler qu'ils ont tous le même objectif : minimiser notre fonction de perte. Adam est généralement considéré comme assez robuste dans la sélection d'hyper paramètres, bien que la vitesse d'apprentissage doive parfois être modifiée par rapport à la valeur par défaut suggérée.

L'algorithme de cette méthode :

**Algorithme** : the ADAM algorithme

**Require**: Step size  $\epsilon$  (Suggested default: 0.001)

**Require**: Exponential decay rates for moment estimates,  $\rho_1$  and  $\rho_2$  in  $[0,1)$ . (Suggested defaults: 0.9 and 0.999 respectively)

**Require**: Small constant  $\delta$  used for numerical stabilization (Suggested default:  $10^{-8}$ )

**Require**: Initial parameters  $\theta$

Initialize 1st and 2nd moment variables  $s = 0, r = 0$

Initialize time step  $t = 0$

**While** stopping criterion not met **do**

Sample a minibatch of  $m$  examples from the training set  $\{x^{(1)} \dots x^{(m)}\}$  with corresponding targets  $y^{(i)}$ .

Compute gradient :  $g \leftarrow \frac{1}{m} \nabla_{\theta} \sum_i L(f(x^i; \theta), y^i)$

$t \leftarrow t + 1$

Update bias first moment estimate:  $s \leftarrow \rho_1 s + (1 - \rho_1)g$

Update bias second moment estimate:  $r \leftarrow \rho_2 r + (1 - \rho_2)g \odot g$

Correct bias in first moment:  $\hat{s} \leftarrow \frac{s}{1 - \rho_1^t}$

Correct bias in second moment:  $\hat{r} \leftarrow \frac{r}{1 - \rho_2^t}$

Compute update: Compute update:  $\Delta\theta \leftarrow -\epsilon \frac{\hat{s}}{\sqrt{\hat{r} + \delta}}$  (operation applied element-wise)

Apply update:  $\theta \leftarrow \theta + \Delta\theta$

**End While**

## 2.7 Conclusion

Dans ce chapitre nous avons défini ce qu'un rayon X ainsi qu'une radiographie, Nous nous sommes intéressés à la radiographie du thorax Pour ensuite comparer la radiographie d'une personne atteinte de coronavirus à celle d'une personne non atteinte, à ce titre nous avons expliqué la relation entre la radiographie X et la COVID-19. On a ensuite présenté le réseau de neurones convolutif qu'on va l'utiliser pour notre système automatique ainsi que ces caractéristiques pour clôturer par la présentation de la base de données utilisée pour notre étude ainsi que la définition d'un optimiseur.

# Chapitre 3

## Résultat de la détection automatique

### 3.1 Introduction

Dans ce dernier chapitre nous allons présenter l'architecture de notre modèle ainsi que les résultats obtenus (numériques et graphiques) après le teste de plusieurs optimiseurs.

### 3.2 L'architecture du modèle utilisé

Le meilleur outil pour un travail de classification d'images est un réseau de neurones convolutif (CNN), ce qu'on va utiliser dans notre modèle de détection de la COVID-19, essayons alors d'en utiliser un sur nos données. Comme nous n'avons que peu d'exemples, notre préoccupation première devrait être l'overfitting, c'est le cas où le modèle rencontre trop peu d'exemples, c'est-à-dire lorsque le modèle commence à utiliser des caractéristiques non pertinentes pour faire des prédictions.

L'augmentation des données est un moyen de lutter contre l'overfitting, mais ce n'est pas suffisant puisque nos échantillons augmentés sont encore fortement corrélés. Pour lutter contre l'overfitting, nous devons nous concentrer sur la capacité entropique de notre modèle, c'est-à-dire sur la quantité d'informations que notre modèle est autorisé à stocker. Un modèle qui peut stocker beaucoup d'informations a le potentiel d'être plus précis en exploitant d'avantage de caractéristiques, mais il risque aussi de commencer à stocker des caractéristiques non pertinentes. Cependant, un modèle qui ne peut stocker que quelques caractéristiques devra se concentrer sur les caractéristiques les plus importantes trouvées dans les données, et celles-ci ont plus de chances d'être vraiment pertinentes et de mieux se généraliser.

Il existe différentes façons de moduler la capacité entropique. La principale est le choix du nombre de paramètres dans notre modèle, c'est-à-dire le nombre de couches et la taille de chaque couche.

Dans notre cas, nous utiliserons un très petit CNN avec peu de couches et peu de filtres par couche, parallèlement à l'augmentation et à l'abandon des données. L'abandon contribue également à réduire l'overfitting, en empêchant une couche de voir deux fois exactement le même modèle, agissant ainsi d'une manière analogue à l'augmentation des données (on pourrait dire que l'abandon et l'augmentation des données ont toutes deux tendances à perturber les corrélations aléatoires qui se produisent dans vos données).

L'extrait de code ci-dessous est notre modèle, un simple empilement de 3 couches de convolution avec une activation Relu et suivi par des couches de max-pooling. Ce modèle est très similaire aux architectures que Yann LeCun a préconisé dans les années 1990 pour la classification des images (à l'exception de Relu).

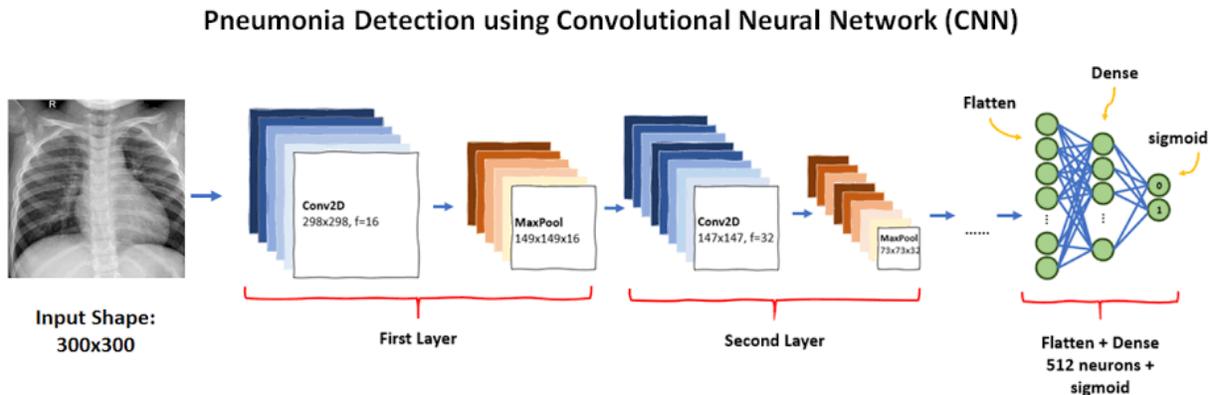


Figure 12 : L'architecture de modèle utilisé.

Pour la première couche on a utilisé 32 filtres avec le (kernel\_size) qui spécifie la taille du filtre (3,3) convolutif en pixels et le input\_shape= (224,224,3) qui divise l'image en 224 pixels avec 3 couleur. Ces derniers nous donnent une sortie (output\_shape) de 222 pixels, après on a utilisé plusieurs filtres pour toutes les couches. La figure suivante nous donne toutes les informations des sorties de notre architecture :

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 222, 222, 32)	896
conv2d_2 (Conv2D)	(None, 220, 220, 128)	36992
max_pooling2d_1 (MaxPooling2D)	(None, 110, 110, 128)	0
dropout_1 (Dropout)	(None, 110, 110, 128)	0
conv2d_3 (Conv2D)	(None, 108, 108, 64)	73792
max_pooling2d_2 (MaxPooling2D)	(None, 54, 54, 64)	0
dropout_2 (Dropout)	(None, 54, 54, 64)	0
conv2d_4 (Conv2D)	(None, 52, 52, 128)	73856
max_pooling2d_3 (MaxPooling2D)	(None, 26, 26, 128)	0
dropout_3 (Dropout)	(None, 26, 26, 128)	0
flatten_1 (Flatten)	(None, 86528)	0
dense_1 (Dense)	(None, 64)	5537856
dropout_4 (Dropout)	(None, 64)	0
dense_2 (Dense)	(None, 1)	65
=====		
Total params: 5,723,457		
Trainable params: 5,723,457		
Non-trainable params: 0		

Tableau 11 : Les paramètres de l'architecture CNN utilisée.

Dans notre architecture, nous collons deux couches entièrement connectées. Nous terminons le modèle avec une seule unité et une activation sigmoïde, ce qui est parfait pour une classification binaire. Nous utiliserons également la perte de cross entropie binaire pour entraîner notre modèle.

### 3.3 Application des optimiseurs

Dans cette étude on va tester les optimiseurs que nous avons présenté dans le chapitre précédent et comparer entre les résultats obtenus pour conclure le meilleur optimiseur.

#### a) Résultat et Discussion de la méthode SGD

On utilise le SGD comme indique la figure suivante :

```
model.compile(loss=keras.losses.binary_crossentropy,  
              optimizer='sgd',  
              metrics=['accuracy'])
```

Figure 13 : Commande python pour la méthode SGD.

Cette méthode nous donne une précision de 0.9667 (96,67%) et une perte de 0.5135 (51,35%).

La méthode du gradient stochastique est un principe plus général, dans lequel la direction de la mise à jour est une variable aléatoire, dont l'espérance est le véritable gradient qui nous intéresse. La SGD peut être beaucoup plus rapide que la méthode du gradient ordinaire (Batch), parce que le nombre de mises à jour est beaucoup plus grand.

La matrice de confusion de cette méthode pour évaluer la précision de classification est donnée dans la figure suivante :

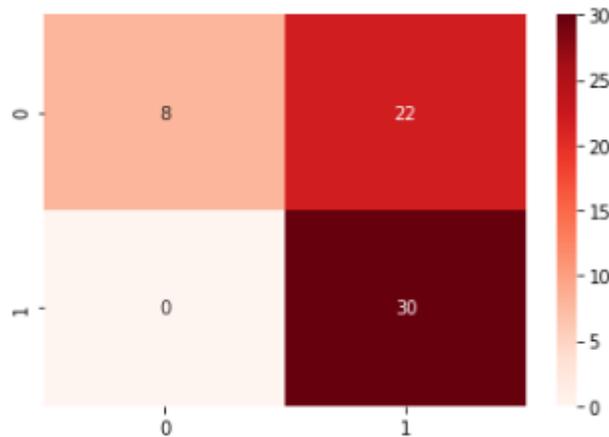


Figure 14 : Matrice de confusion de SGD

On remarque d'après cette matrice que notre modèle a fait une confusion dans plusieurs cas, ce qui confirme que cette méthode n'est pas la bonne pour notre modèle.

Les taux d'entraînement et de validation pour l'optimiseur 'SGD' sont présentés dans les figures suivantes pour la précision (accuracy) et la perte (loss) :

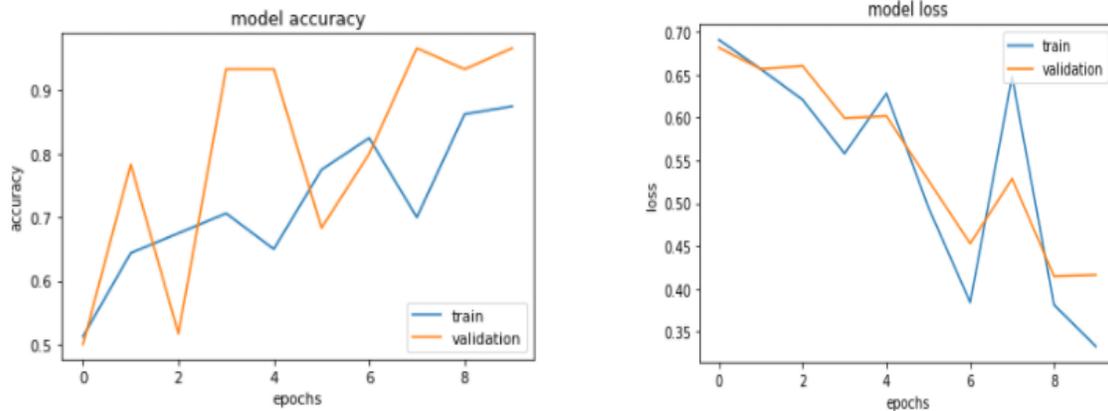


Figure 15 : Les taux d'entraînement et de validation pour la méthode SGD

On remarque pour les deux graphes que : la précision (accuracy) dans la validation est plus élevée que dans l'entraînement et que la perte (loss) dans la validation est plus basse que dans l'entraînement ce qui est bon signe, par contre les deux graphes ne sont pas stables à cause des époques d'entraînement qu'on a fait, pour cela on doit augmenter le nombre d'époques mais nous n'avons pas la capacité nécessaire pour le faire (capacité du matériel « PC »).

## b) Résultats et Discussion de la méthode RMSprop

```
model.compile(loss=keras.losses.binary_crossentropy,  
              optimizer='rmsprop',  
              metrics=['accuracy'])
```

Figure 16 : Commande python pour la méthode RMSprop.

Cette méthode nous donne une précision de 0.9833 (98,33%) et une perte de 0.2405 (24,05%).

Cet algorithme RMSprop est une version alternative d'AdaGrad et ressemble à de la méthode du gradient avec Momentum (la différence se fait au niveau du calcul mathématique du gradient). Cette méthode converge plus rapidement que le SGD comme on a dit précédemment,

c'est grâce à cette rapidité qu'on remarque la grande différence dans les pertes entre les deux méthodes.

La matrice de confusion de cette méthode pour évaluer la précision de classification est donnée dans la figure suivante :

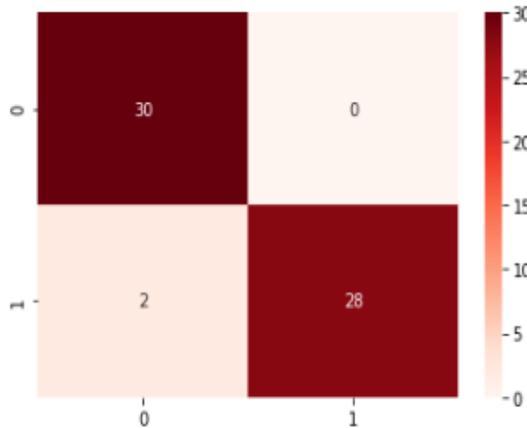


Figure 17 : Matrice de confusion de RMSprop

On remarque d'après cette matrice que notre modèle a fait une confusion dans deux cas seulement, ce qui confirme le bon résultat de cette méthode.

Les taux d'entraînement et de validation pour l'optimiseur 'RMSprop' sont présentés dans les figures suivantes pour la précision (accuracy) et la perte (loss) :

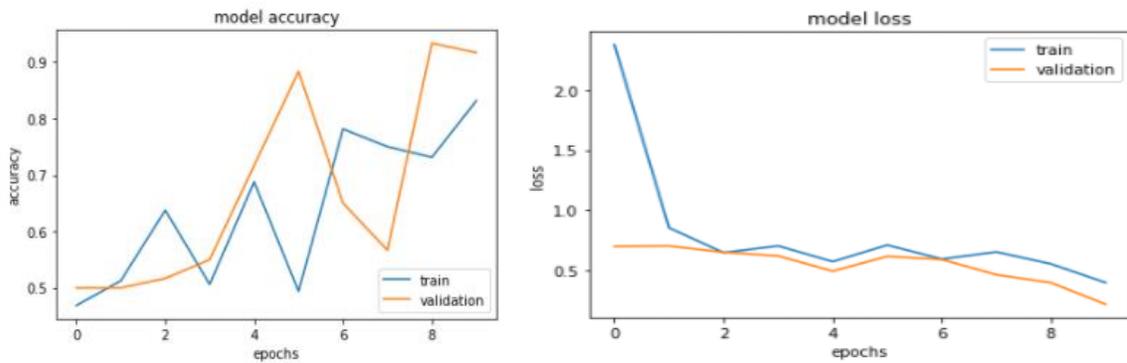


Figure 18 : Les taux d'entraînement et de validation pour la méthode RMSprop

On remarque pour les deux graphes que : la précision (accuracy) dans la validation est plus élevée que dans l'entraînement et que la perte (loss) dans la validation est plus basse que dans l'entraînement ce qui est bon signe, par contre les deux graphes ne sont pas stables à cause des époques d'entraînement qu'on a fait, pour cela on doit augmenter le nombre d'époques mais nous n'avons pas la capacité nécessaire pour le faire (capacité du matériel « PC »).

### c) Résultats et Discussion de la méthode Adam

On utilise Adam comme indique la figure suivante :

```
model.compile(loss=keras.losses.binary_crossentropy,  
              optimizer='adam',  
              metrics=['accuracy'])
```

Figure 19 : Commande python pour la méthode Adam.

Cette méthode nous donne une précision de 0.9667 (96,67%) et une perte de 0.1043 (10,43%).

Il s'agit de l'optimiseur le plus utilisé (surtout pour les réseaux de neurones) en raison de son efficacité et de sa stabilité. Pour Adam, le gradient est dans la même direction que pour les algorithmes précédents, on va accélérer la vitesse (dite d'apprentissage) à laquelle on descend la courbe (i.e. on met à jour les paramètres). A noter qu'Adam combine les idées d'AdaGrad et de RMSprop qui son basé sur le SGD donc elle nous donne les meilleurs résultats.

La matrice de confusion de cette méthode pour évaluer la précision de classification est donnée dans la figure suivante :

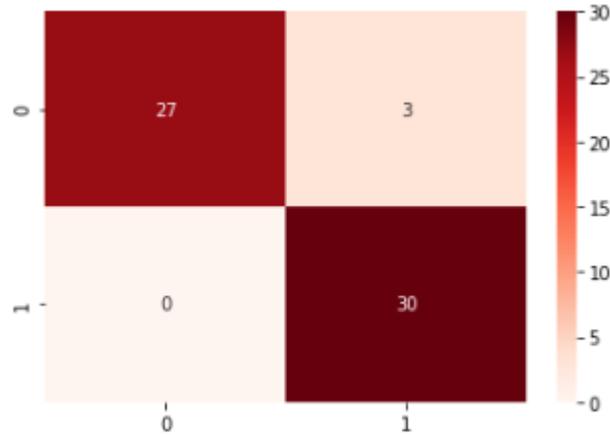


Figure 20 : Matrice de confusion d'Adam

On remarque d'après cette matrice que notre modèle a fait une confusion dans trois cas seulement, ce qui confirme aussi les bons résultats de cette méthode.

Les taux d'entraînement et de validation pour l'optimiseur 'Adam' est présenté dans les figures suivantes pour la précision (accuracy) et la perte (loss) :

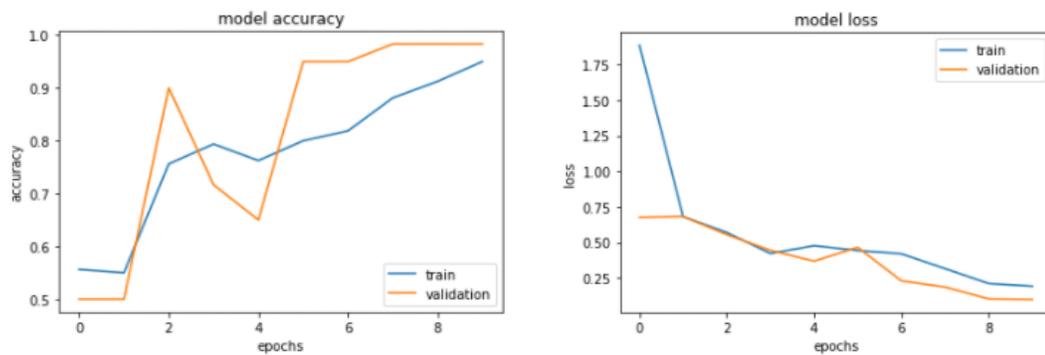


Figure 21 : Les taux d'entraînement et de validation pour la méthode Adam

On remarque pour les deux graphes que : la précision (accuracy) dans la validation est plus élevée que dans l'entraînement et que la perte (loss) dans la validation est plus basse que dans l'entraînement ce qui est bon signe, par contre les deux graphes ne sont pas stables à cause des époques d'entraînement qu'on a fait, pour cela on doit augmenter le nombre d'époques mais nous n'avons pas la capacité nécessaire pour le faire (capacité du matériel « PC »).

#### d) Tableau des résultats

On résume les résultats obtenus dans un tableau :

Méthode utilisé	Accuracy (Précision)	Loss (perte)	Commentaires
SGD	0.9667 (96,67%)	0.5135 (51,35%)	C'est la méthode qui donne les moins bons résultats car c'est la plus ancienne et elle converge moins rapidement que les autres méthodes.
RMSprop	0.9833 (98,33%)	0.2405 (24,05%)	RMSprop donne des très bons résultats avec une meilleure valeur de précision.
Adam	0.9667 (96,67%)	0.1043 (10,43%)	C'est la meilleure méthode entre les trois avec la meilleure valeur de perte.

Tableau 12 : Le résumé des résultats

### 3.4 Discussion

Nous avons testé plusieurs optimiseurs dans cette partie qui nous ont donné différents résultats. Le SGD a donné une Précision de 0.9667 et une perte de 0.5135, comme on peut remarquer,

cette méthode donne les moins bons résultats du fait qu'elle soit ancienne et qu'elle fasse le moins de mise à jour par rapport aux autres méthodes. RMSprop c'est la deuxième méthode qu'on a testée et qui a donné une Précision de 0.9833 et une perte de 0.2405, on remarque clairement la différence entre le SGD et le RMSprop qui a donné des bons résultats grâce à la différence de calcul qui cause une convergence plus rapide. Adam est la dernière méthode qu'on a appliquée et qui nous a donné les meilleurs résultats avec une Précision de 0.9667 et une perte de 0.1043, on remarque une petite différence entre Adam et RMSprop dans la précision et une grande différence dans les pertes. Adam est la méthode la plus convenable et utilisée avec les CNN, elle combine aussi plusieurs méthodes ce qui donne une convergence plus rapide et de meilleurs résultats.

### **3.5 Conclusion**

Dans ce chapitre nous avons présenté l'architecture du modèle de détection de la COVID-19. Nous avons appliqué plusieurs optimiseurs -antérieurement présentés- sur le modèle de détection de la COVID-19 et nous en avons conclu à l'aide des résultats et des graphes qu'Adam est la meilleure méthode.

# Conclusion

La détection rapide de la COVID-19 avec un système automatique est très importante et primordiale pour gagner du temps et lutter contre cette pandémie. Nous avons appliqué un programme python et tester plusieurs optimiseurs qui ont donnée de bons résultats, et on a conclue qu'Adam est la meilleure méthode.

Un réseau de neurones convolutif est un système composé de neurones, généralement répartis en plusieurs couches connectées entre elles, Un tel système s'utilise pour résoudre divers problèmes statistiques, mais nous nous intéressons ici qu'au problème de classification (très courant). Dans le cas de cette étude, le réseau classe les radiographie selon deux catégories : atteint de COVID-19 ou sain.

Nous avons utilisé différents optimiseurs en précisant les critères de comparaison entre les résultats obtenus. Cependant il existe d'autre optimiseurs que nous n'avons ni évoqués ni utilisés tels qu'AdaGrad et Momentum et qui donnent d'assez bons résultats.

La porte de la recherche sur les applications de l'intelligence artificielle est toujours ouverte à de nouveaux algorithmes et propositions. Nous pensons que les ingénieurs doivent aller plus loin dans ce domaine et doivent acquérir davantage de connaissance afin de mettre en place plus de théories et de les convoiter.

# Bibliographie

- [1] Alexandra Lapierre, Guillaume Fontaine, Pierre-Luc Tremblay, Marc-André Maheu-Cadotte et Michaël Desjardins : La maladie à coronavirus (COVID-19) : portrait des connaissances actuelles.

URL :

- [https://www.researchgate.net/profile/Alexandra\\_Lapierre/publication/341355504\\_La\\_maladie\\_a\\_coronavirus\\_COVID-19\\_portrait\\_des\\_connaissances\\_actuelles/links/5ebc1bdb92851c11a8675603/La-maladie-a-coronavirus-COVID-19-portrait-des-connaissances-actuelles.pdf](https://www.researchgate.net/profile/Alexandra_Lapierre/publication/341355504_La_maladie_a_coronavirus_COVID-19_portrait_des_connaissances_actuelles/links/5ebc1bdb92851c11a8675603/La-maladie-a-coronavirus-COVID-19-portrait-des-connaissances-actuelles.pdf)
- [2] Kim, P. (2017). Matlab deep learning. *With Machine Learning, Neural Networks and Artificial Intelligence*, 130, 21.
- [3] Chollet, F. (2018). Deep Learning mit Python und Keras: Das Praxis-Handbuch vom Entwickler der Keras-Bibliothek. MITP-Verlags GmbH & Co. KG.
- [4] Narin, A., Kaya, C., & Pamuk, Z. (2020). Automatic detection of coronavirus disease (COVID-19) using x-ray images and deep convolutional neural networks. arXiv preprint arXiv:2003.10849.
- [5] Apostolopoulos, I. D., & Mpesiana, T. A. (2020). COVID-19: automatic detection from x-ray images utilizing transfer learning with convolutional neural networks. *Physical and Engineering Sciences in Medicine*, 1.
- [6] Saeedi, A., Saeedi, M., & Maghsoudi, A. (2020). A novel and reliable deep learning web-based tool to detect COVID-19 infection form chest CT-scan. arXiv preprint arXiv:2006.14419.
- [7] Heidari, M., Khuzani, A. Z., Hollingsworth, A. B., Danala, G., Mirniaharikandehei, S., Qiu, Y., ... & Zheng, B. (2018). Prediction of breast cancer risk using a machine learning approach embedded with a locality preserving projection algorithm. *Physics in Medicine & Biology*, 63(3), 035020.
- [8] Sun, S., Cao, Z., Zhu, H., & Zhao, J. (2019). A survey of optimization methods from a machine learning perspective. *IEEE transactions on cybernetics*.

- [9] TAFFAR, M. INITIATION AL'APPRENTISSAGE.
- [10] Shalev-Shwartz, S., & Zhang, T. (2013). Stochastic dual coordinate ascent methods for regularized loss minimization. *Journal of Machine Learning Research*, 14(Feb), 567-599.
- [11] La radiographie ET L'imagerie par rayons X.  
URL: <https://parlonssciences.ca/ressources-pedagogiques/documents-dinformation/la-radiographie-et-limagerie-par-rayons-x>
- [12] Radiographie thoracique  
URL: <https://www.ottawaheart.ca/fr/examen-intervention/radiographie-thoracique>
- [13] Roosa, K., Lee, Y., Luo, R., Kirpich, A., Rothenberg, R., Hyman, J. M., ... & Chowell, G. (2020). Real-time forecasts of the COVID-19 epidemic in China from February 5th to February 24th, 2020. *Infectious Disease Modelling*, 5, 256-263.