

الجمهورية الجزائرية الديمقراطية الشعبية
وزارة التعليم العالي والبحث العلمي

UNIVERSITÉ BADJI MOKHTAR - ANNABA
BADJI MOKHTAR – ANNABA UNIVERSITY



جامعة باجي مختار – عنابة

Faculty: Engineering Sciences

Département : Informatique

Domaine : Mathématiques et Informatique

Filière : Informatique

Spécialité : intelligence artificielle

Mémoire

Présenté en vue de l'obtention du Diplôme de Master

Thème:

Un système Deep Learning pour la Classification de visages portant un masque .

Présenté par : Farah Aymen

Jury de Soutenance :

Djebbar Akila	MCB	Université Badji Mokhtar-Annaba	Président
Amirouche Radia	MAA	Université Badji Mokhtar-Annaba	Encadrant
Kobsi Noha	MAA	Université Badji Mokhtar-Annaba	Examineur

Année Universitaire : 2020/2021

« L'avenir d'un enfant est l'œuvre de sa mère »

Napoléon Bonaparte.

Résumé

[The overall objective of this project is the design of an effective classifier that can detect whether a person is wearing a medical mask or not. To this end, an easy and accurate face detector that can extract ROI using input from a camera setup must be designed and deployed. The implemented solution uses OpenCv's haar cascade, then the extracted image is presented to a pre-trained CNN classifier and transferred for our case study.]

L'objectif général de ce projet est la conception d'un classifieur efficace qui peut bien détecter si une personne porte un mask ou pas . À cette fin, un détecteur de visage facile et précis qui peut extraire le ROI en utilisant l'entrée d'une configuration de caméra doit être conçu et déployé. La solution mise en œuvre utilise la cascade haar d'OpenCv, puis l'image extraite est présentée à un classificateur CNN déjà entraîné. ...

Dédicace

Maman merci d'avoir toujours été là pour moi! Tu es restée ma plus belle certitude quand tout le reste s'est effondré. Tu étais là : constante et forte, inébranlable, ma consolatrice, mon réconfort, patiente, indulgente, bienveillante et compréhensive, la meilleure des amies.

Tu es mon réconfort et ma certitude. Mon guide dans l'existence Mon courage sans rage. Un compagnon de vie si sage. Depuis ma naissance ma plus belle chance.

Pour mon encadrant Dr Radia Amirouche : Merci à vous pour votre aide et vos pensées qui me donnent beaucoup d'espérance. Mille mercis pour votre belle gentillesse et votre disponibilité Votre soutien et aide témoignent de votre grande générosité Vous avoir croisé dans mon chemin est l'une des plus belles choses qui m'est arrivé.

Table de Matière

Résumé	ii
Dédicace	iii
1 Introduction	1
2 Recherche	2
2.1 Objectif	2
2.2 Travaux relatifs	2
2.3 Questions de recherche	3
2.4 Méthodes de recherche	4
3 Réseaux de neurones et apprentissage profond	6
3.1 Introduction au chapitre	6
3.2 Apprentissage automatique	6
3.3 Apprentissage profond	8
3.4 Réseau de neurones artificiel	9
3.5 Réseau de neurones convolutif	10
3.5.1 Composent du CNN :	12
Couche d'entrée :	12
Convolution/filtering layer :	12
Couche de mise en commun :	13
Couche de correction linéaire (ReLU) :	14
Aplatissement :	14
Couche de sortie (Dense) :	15
3.5.2 Quelques architectures CNN célèbres :	15
AlexNet :	15
VGG16 :	16
VGG19 :	16
ResNet :	16
3.6 Conclusion du chapitre	17
4 Conception	18
4.1 Introduction au chapitre	18
4.2 Notre modele convolutif	18
4.3 Comparaisons de différents modeles	18
4.4 Pré-traitement :	19
4.5 L'architecture CNN :	21
4.6 Conclusion du chapitre	23

5	Jeu de données et outils utilisés	24
5.1	Introduction au chapitre	24
5.2	Data Set : with-without-mask	24
5.3	Outils et environnements utilisés :	26
5.3.1	Google Collaboratory :	26
5.3.2	OpenCV :	26
5.3.3	Keras :	27
5.3.4	TensorFlow :	27
6	Discussion sur les résultats	28
6.1	introduction du chapitre	28
6.2	Evaluation du CNN	29
6.2.1	Perte (Loss)	29
	Fonction de perte (Loss function)	30
	Graphe de perte	30
6.2.2	Précision (Accuracy)	31
6.2.3	Matrice de Confusion	32
	Conclusion	33
	Bibliographie	34

Table de Figures

3.1	Ordre hiérarchique de l'IA, du ML et de l'apprentissage profond	8
3.2	Représentation schématique d'une couche de convolution dans un CNN.[5].	12
3.3	Différence entre les types de pooling	13
3.4	the Flattening layer	14
3.5	Alex-net architecture	15
3.6	Vgg16 architecture	16
4.1	19
4.2	Comparaison des performances des modeles	20
4.3	extraction du visage	20
4.4	l'architecture mobile net	22
4.5	Diagramme de l'architecture	22
5.1	échantillons des données	25
5.2	Distrubition des données	25
6.1	Test du modèle	28
6.2	Epoch Loss	30
6.3	Epoch Accuracy	31
6.4	Matrice de confusion	32

Chapitre 1

Introduction

Le COVID-19 peut se propager par contact et par des surfaces contaminées. Par conséquent, les systèmes biométriques classiques basés sur les mots de passe ou les empreintes digitales ne sont plus sûrs. La reconnaissance faciale est plus sûre, sans qu'il soit nécessaire de toucher un quelconque dispositif. Des études récentes sur le coronavirus ont prouvé que le port d'un masque facial par une population saine et infectée réduit considérablement la transmission de ce virus. Cependant, le port du masque facial pose les problèmes suivants :

- Les fraudeurs et les voleurs profitent du masque pour voler et commettre des délits sans être identifiés.
- Le contrôle d'accès communautaire et l'authentification du visage sont devenus des tâches très difficiles lorsqu'une grande partie du visage est cachée par un masque.
- Les méthodes existantes de reconnaissance des visages ne sont pas efficaces lorsque l'on porte un masque qui ne peut fournir l'image complète du visage pour la description.
- L'exposition de la région du nez est très importante dans la tâche de reconnaissance des visages car elle est utilisée pour la normalisation du visage, la correction de la pose et la correspondance des visages.

En raison de ces problèmes, les masques de visage ont mis au défi les méthodes existantes de reconnaissance des visages.

Pour résoudre ces problèmes, nous distinguons deux tâches différentes, à savoir : la reconnaissance des masques de visage et la reconnaissance des visages masqués. La première tâche consiste à vérifier si la personne porte un masque ou non. Elle peut être appliquée dans les lieux publics où le masque est obligatoire. La reconnaissance de visages masqués, quant à elle, vise à reconnaître un visage avec un masque en se basant sur les régions des yeux et du front. Dans ce travail, nous traitons la première tâche en utilisant une méthode basée sur l'apprentissage profond.

Le reste de cet article est organisé comme suit : Le chapitre 1 présente les travaux connexes. Dans le chapitre 2, nous présentons la motivation et la contribution de l'article. La méthode proposée est détaillée dans le chapitre 3. Les résultats expérimentaux sont présentés dans le chapitre 4. La conclusion termine le document.

Chapitre 2

Recherche

2.1 Objectif

La détection d'objets est l'un des problèmes de la vision par ordinateur qui a fait l'objet d'un grand nombre d'études. Au cours de la décennie précédente. De nombreux chercheurs ont contribué à développer et d'innover de nouvelles théories pour améliorer la vision par ordinateur et d'appliquer ces théories en diverses études qui ont été publiées sur sujet et qui se concentre sur développer des méthodes permettant de vérifier la présence d'un visage et de le reconnaître. L'étude mène pour clarifier les obstacles à la détection et à l'identification et introduire une explication qui permet de développer une nouvelle méthode de système de détection à la fois pour notre cas d'étude.

2.2 Travaux relatifs

En général, la plupart des publications se concentrent sur la construction du visage et la reconnaissance de l'identité lors du port de masques. Dans cette recherche, nous nous concentrons sur la reconnaissance des personnes qui ne portent pas de masque facial pour aider à réduire la transmission et la propagation du COVID-19.

Les chercheurs et les scientifiques ont prouvé que le port de masques aide à minimiser le taux de propagation du COVID-19. Dans [QIN and D. Li, Identifying facemask-wearing condition using image super-resolution with classification network to prevent COVID-19, May 2020], les auteurs ont développé une nouvelle méthode d'identification des conditions de port du masque. Ils ont été capables de classer trois catégories de conditions de port de masque. Ces catégories sont le port correct du masque, le port incorrect du masque et l'absence de port du masque. La méthode proposée a atteint une précision de 98,70 dans la phase de détection des visages. [S. Ejaz, M.R. Islam, M. Sifatullah, A. Sarker, Implementation of principal component analysis on masked and non-masked face recognition in : 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT), 2019], ont appliqué l'analyse en composantes principales (PCA) à la reconnaissance des visages masqués et non masqués pour reconnaître les personnes. Ils ont trouvé que la précision de la reconnaissance des visages en utilisant l'ACP est extrêmement affectée par le port de masques. La précision de la reconnaissance tombe à moins de 70 lorsque le visage reconnu

est masqué. L'ACP a également été utilisée dans [Jeong-Seon Park, You Hwa Oh, Sang ChulAhn, and Seong-Whan Lee, Glasses removal from facial image using recursive error compensation, IEEE Trans. Pattern Anal. Mach. Intell. 27 (5) (2005)]. Les auteurs ont proposé une méthode utilisée pour enlever les lunettes d'une image faciale frontale humaine. La partie enlevée a été reconstruite en utilisant une compensation d'erreur récursive utilisant la reconstruction PCA. Dans [12], les auteurs ont utilisé l'algorithme YOLOv3 pour la détection des visages. YOLOv3 utilise Darknet-53 comme colonne vertébrale. La méthode proposée a atteint une précision de 93,9. Elle a été entraînée sur les bases de données CelebA et WIDER FACE comprenant plus de 600 000 images. Le test a été effectué sur le jeu de données FDDB. Nizam et al [13] ont proposé un nouveau réseau basé sur un GAN qui peut automatiquement supprimer les masques couvrant la zone du visage et régénérer l'image en construisant le trou manquant. Le résultat du modèle proposé est une image complète du visage qui semble naturelle et réaliste. Dans [14], les auteurs ont présenté un système de détection de la présence ou de l'absence d'un masque médical obligatoire dans une voiture. Absence d'un masque médical obligatoire dans la salle d'opération. L'objectif global de L'objectif global est de minimiser les détections de visage faussement positives sans manquer les détections de masques afin de déclencher des alarmes seulement pour le personnel médical qui ne porte pas de masque chirurgical. Le système proposé a obtenu une précision de 95 Muhammad et al [15] ont présenté une méthode interactive appelée MRGAN. La méthode repose sur l'obtention de la zone du microphone de la part de l'utilisateur et sur l'utilisation du réseau additif génératif pour reconstruire cette zone. L'utilisateur et l'utilisation du Generative Adversarial Network pour reconstruire cette zone. Shaik et al [16] ont utilisé l'apprentissage profond pour la classification et la reconnaissance des et la reconnaissance des émotions faciales en temps réel. Ils ont utilisé le VGG-16 pour classifier sept expressions faciales. Le modèle proposé a été entraîné sur le jeu de données KDEF et a obtenu une précision de 88 d'exactitude.

2.3 Questions de recherche

Selon Dawson R. Hancock et Bob Algozzine (2017, 3), divers types de questions (Quoi ? Comment ? Pourquoi ?) qui ont poussé les chercheurs à explorer les causes pour lesquelles les choses se sont produites et à créer des approches plus spécifiques. Habituellement, lorsque des spécialistes analysent un sujet, cela implique qu'ils cherchent à obtenir des retours d'information pour une meilleure compréhension du sujet, des scénarios alternatifs pour l'analyse, et des explications possibles à examiner. Ces retours d'information et ces questions ont motivé l'étude pour tirer des conclusions qui soient fiables, pratiques et interprétables. Selon Dawson H. et al. (2017, 4), un effort de recherche ne devrait pas être mené par un chercheur sans paradigme organisationnel. Un chercheur sans paradigme organisationnel. Ce paradigme définit

pour le chercheur les caractéristiques distinctives de l'étude et les possibilités pour obtenir des réponses aux questions. Par conséquent, l'auteur détermine un paradigme pour mener une recherche systématique sur le sujet de l'étude en identifiant trois piliers essentiels pour concevoir le processus de recherche. Piliers essentiels pour concevoir le processus de recherche : - Les méthodes de l'étude - Collecte d'informations - La confirmation des résultats. Ces trois piliers essentiels ont conduit l'étude à comprendre les méthodes de recherche, les données, données et les formulaires d'analyse pour illustrer les étapes de l'OFR et comprendre les données requises pour construire un modèle très précis. En suivant les étapes précédentes, les questions de recherche basées sur le triple paradigme sont :

- Quel est la meilleure approche pour développer un tel système ?
- Comment la quantité et la qualité des données va affecter les performances de notre système ?
- Quel sont les métriques pour évaluer les performances de notre système ?

2.4 Méthodes de recherche

En répondant à une question telle que "Qu'est-ce que cette étude essaie de faire ? des méthodes de recherche spécifiques pour trouver la réponse. La réponse à cette question et l'objectif de cette étude est de trouver une nouvelle approche pour résoudre le problème des RFO.

Par conséquent, l'étude a commencé à mener des recherches quantitatives en mettant en place des expériences contrôlées et en collectant des données par le biais de différentes ressources, des expériences contrôlées et en collectant des données par le biais de différentes ressources. Selon Claes Wohlin, Martin Höst, et Kennet Henningsson (2003,2) "les données quantitatives favorisent les comparaisons et l'analyse statistique". Et en utilisant la littérature disponible aux différents contributeurs de la société de vision par ordinateur tels que l'Université de Stanford, les laboratoires de recherche Google AI, l'Institut de technologie du Massachusetts, et les travaux de nombreux chercheurs. L'auteur a été en mesure de recueillir des données essentielles pour essentielles à utiliser dans la recherche pour améliorer les performances du système en comparant et en analysant les résultats. L'étude a considéré que les matériaux collectés étaient dignes d'intérêt, en particulier le travail de Gregory Koch (2015) dans sa thèse "Siamese Neural Networks pour la reconnaissance d'images à un coup", où Koch a exploré une méthode pour classifier la similarité entre deux figures. Et la thèse d'Ali Sharif Razavian(2017) dans "Convolutional Network Representation for Visual Recognition" qui a décrit la représentation du réseau convolutif en reconnaissance visuelle d'un point de vue empirique.

L'étude fournit des informations générales vérifiables qui serviront de base à la mise en œuvre du système OFR en temps réel et à la résolution des problèmes. Mettre en œuvre le système OFR en temps réel et relever les défis. Le site recherche se concentre sur plusieurs points de vue différents qui se combinent pour répondre à questions, clarifier les méthodes de travail et les

mécanismes utilisés pour développer les algorithmes, et étudier les recommandations. L'étude utilise des expériences, ce qui nécessite plusieurs variables pour les mener et en examiner les effets. Ces variables, telles que les données, les hypothèses, les examens, les performances des appareils et les circonstances environnementales sont plus importantes dans la recherche que l'expérience elle-même. Deux méthodes de recherche quantitative ont été appliquées pour mener les expériences et analyser les résultats.

D'après Claes Wohlin, Martin Höst, et Kennet Henningsson (2003, 9), la méthode de recherche empirique peut être décrite comme suit : étapes suivantes : Définition, Planification, Exploitation, Analyse et Conclusion. L'objectif de la recherche empirique est de manipuler une ou plusieurs variables et de contrôler toutes les autres variables à un niveau fixe. Et l'effet de la manipulation des variables peut être mesuré sur la base de résultats statistiques. Étant donné que nous comparons différents types de méthodes dans chaque étape et que nous analysons les résultats pour calculer les ratios, la méthode de recherche empirique est le modèle approprié pour cette recherche quantitative. Les données utilisées pour cette étude ont été collectées à partir de différentes ressources (livres, articles, revues, l'expérience accumulée par l'auteur) ont été obtenues en utilisant des mots-clés spécifiques tels que : Intelligence artificielle, Réseau neuronal, Apprentissage automatique, Apprentissage profond, Vision informatique, Vision par ordinateur, détection et localisation d'objets, reconnaissance et comparaison d'objets, Examens d'algorithmes, et enfin ensembles de données.

La méthode de recherche empirique permet d'obtenir des preuves et d'observer les données scientifiques des expériences, puis d'examiner ces résultats en utilisant la méthode de recherche de l'analyse des données secondaires. Le processus empirique commence par réagir aux préoccupations de recherche - préoccupations de la recherche qui doivent être étudiées afin de déterminer l'orientation de la recherche en mettant en évidence les objectifs fondamentaux de l'enquête systématique.

La deuxième étape consiste à réanalyser les données collectées précédemment et à les comparer aux résultats. L'auteur utilise donc l'analyse secondaire des données qui est une technique de collecte de données largement utilisée dans la recherche scientifique. Selon Melissa P. Johnston (2014,8) " Les principaux avantages associés à l'analyse secondaire sont le rapport coût-efficacité et la commodité qu'elle procure ".

L'analyse secondaire est la meilleure technique pour recueillir des données à plusieurs fins, notamment pour offrir des possibilités de validation pour les répliques, ce qui est substantiel dans l'étude car les résultats de l'étude sont crédibles s'ils se retrouvent dans une variété d'autres études. L'auteur doit considérer comment les données sont catégorisées, organisées, et comment cela peut affecter les résultats. Il doit également apporter les ajustements nécessaires aux données afin de mener l'étude de la bonne manière. Les méthodes choisies apportent des considérations sur les résultats et les évaluations pour mettre en œuvre de nouvelles expériences et déterminer le processus de recherche.

Chapitre 3

Réseaux de neurones et apprentissage profond

3.1 Introduction au chapitre

La recherche dans le domaine de l'intelligence artificielle s'est considérablement développée en plus d'un demi-siècle, en grande partie grâce aux progrès récents de l'apprentissage automatique, et en particulier de l'apprentissage profond, ainsi qu'à l'amélioration radicale de la puissance de calcul du matériel. Ce domaine de recherche a de plus en plus attiré l'attention de nombreuses disciplines et de chercheurs d'horizons très divers et est devenu un sujet largement interdisciplinaire. Avec le développement de la technologie, l'intelligence artificielle s'est répandue et développée rapidement, et a également favorisé le développement d'autres disciplines. Ce chapitre fournit une description des techniques avancées d'apprentissage automatique, avec un accent particulier sur l'apprentissage profond, les réseaux neuronaux, les réseaux de neurones et les réseaux de communication.

3.2 Apprentissage automatique

L'apprentissage automatique est un sous-domaine du groupe plus large des sciences informatiques [13]. L'étude de l'apprentissage automatique est une analyse de données qui automatise la construction de modèles analytiques. L'étude de l'apprentissage automatique est une analyse de données qui automatise la construction de modèles analytiques et, dans un certain sens, l'informatique répond à l'induction. Le site L'apprentissage automatique est une analyse de données qui automatise la construction de modèles analytiques. pas destinés à l'interprétation humaine. Dans l'apprentissage automatique, le résultat est (généralement) de mettre continuellement à jour les actions du programme en conséquence. Comme le suggère le nom "Machine Le domaine est divisé en deux sous-groupes d'apprentissage : supervisé et non supervisé. Cette thèse Cette thèse explore principalement l'apprentissage supervisé, qui consiste à apprendre à partir de données d'entraînement étiquetées et à l'appliquer à des données inconnues. données d'entraînement étiquetées et de l'appliquer à des données inconnues pour faire

des prédictions. Les algorithmes non supervisés peuvent tirer des conclusions à partir d'ensembles de données sans étiquettes pour effectuer le regroupement, la sélection de caractéristiques, etc. [1].

3.3 Apprentissage profond

L'apprentissage profond est une catégorie de techniques d'apprentissage automatique, où de nombreuses couches d'étapes de traitement de l'information dans des architectures hiérarchiques supervisées sont exploitées pour l'apprentissage non supervisé de caractéristiques et pour l'analyse/classification de modèles. L'essence de l'apprentissage profond consiste à calculer des caractéristiques ou des représentations hiérarchiques des données d'observation, où les caractéristiques ou les facteurs de niveau supérieur sont définis à partir de ceux de niveau inférieur.[2].

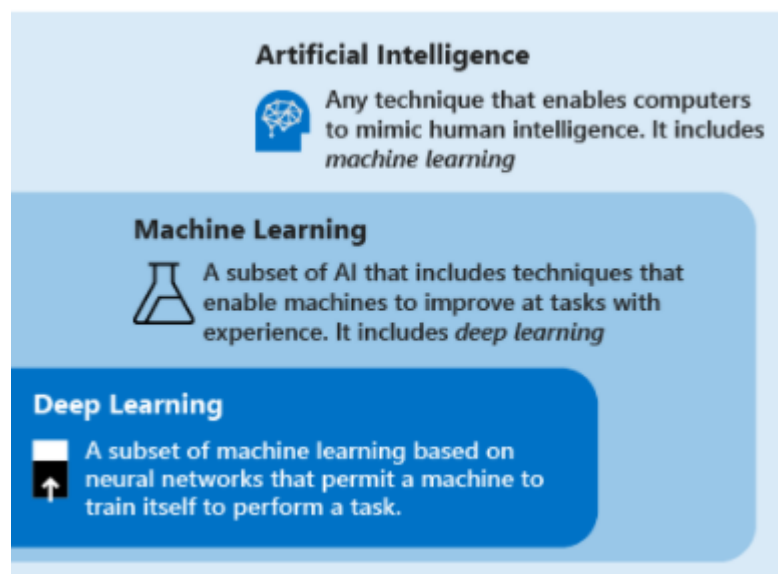


FIGURE 3.1 – Ordre hiérarchique de l'IA, du ML et de l'apprentissage profond

3.4 Réseau de neurones artificiel

Même si les ordinateurs sont conçus par et pour les humains, il est clair que le concept d'un ordinateur est extrêmement différent de celui du cerveau d'une personne. Le cerveau humain peut être complexe et non linéaire et, en outre, sa façon de traiter l'information est très parallélisée. Il repose sur des composants structurels appelés neurones, qui sont tous conçus pour effectuer certaines formes de calculs. Il est appliqué à un très grand nombre de tâches de reconnaissance, qu'il exécute fréquemment en 100 à 200 ms. Les tâches de ce type sont encore très difficiles à traiter et, il y a quelques années encore, effectuer ces calculs sur un processeur pouvait prendre des jours [3]. Inspiré par ce système étonnant, afin de former des ordinateurs plus adaptés à ce type de tâches, une nouvelle façon de traiter ces problèmes a vu le jour. On l'appelle un réseau neuronal artificiel (ANN). Un ANN peut être un modèle supportant un réseau interconnecté potentiellement massif d'unités de traitement, convenablement appelées neurones. Pour que le réseau et ses neurones comprennent la manière de traiter les informations entrantes, le modèle doit acquérir des connaissances. Cela se fait souvent par le biais d'un processus d'apprentissage. Les connexions entre les neurones du réseau sont représentées par des poids, et ces poids stockent les connaissances acquises par le modèle. Ce type de structure conduit à une généralisation élevée et, par conséquent, le fait indéniable que la façon dont les neurones traitent les données sera non linéaire est utile pour toute une série d'applications diverses. Cela révèle des approches totalement nouvelles pour la mise en correspondance des entrées et des sorties et permet la création de modèles de calcul hautement adaptatifs. [3].

3.5 Réseau de neurones convolutif

Les réseaux neuronaux convolutifs sont apparus bien plus tard que les réseaux neuronaux ordinaires, mais ils en partagent encore de nombreuses caractéristiques. Ils accueillent des neurones avec des poids et des biais apprenables qui traitent les données et les font éventuellement passer par des fonctions non linéaires. Ils utilisent également une fonction de perte pour calculer les erreurs de calcul qui seront propagées dans le réseau. Cependant, comme le fichier d'entrée d'un réseau neuronal quotidien est mis à l'échelle, par exemple, d'une image, le réseau très étendu reliant tous les neurones a tendance à devenir ingérable. La structure d'un réseau neuronal convolutif est explicitement conçue pour gérer ce problème, et elle est soutenue par une collection de types de couches spécialisés dans cette tâche. [4]. La principale caractéristique est la couche dite convolutive. Cette couche accepte une entrée tridimensionnelle, typiquement une image avec une hauteur $H1$, une largeur $W1$ et une variété de canaux (ou profondeur) $D1$. Elle consiste en une collection de K filtres tridimensionnels dont les côtés correspondent à la taille du filtre (ou taille du noyau) F et donc à la même profondeur que l'entrée, $D1$. Les filtres sont essentiellement des ensembles de neurones avec des poids et un biais qui traitent les données comme dans un réseau neuronal ordinaire. Pour produire une sortie, chaque filtre balaie l'image d'entrée et, pour chaque étape, produit une valeur à la sortie. La façon dont le filtre parcourt l'image est déterminée par le pas S , qui est la distance entre deux points de la grille de couverture de l'image du filtre. Afin d'approcher les bords de l'image, un autre paramètre P détermine la quantité de remplissage de zéro, c'est-à-dire le nombre de couches de valeurs nulles ajoutées autour de l'image qui seront également incluses dans le balayage. De cette façon, la taille de la sortie peut être égale à celle de l'entrée, quelle que soit la taille du noyau.

Une fois que tous les filtres de la couche ont parcouru l'image en entrée, les sorties bidimensionnelles de tous les filtres sont empilées pour créer le volume de sortie de dimensions :

$$\text{width} : W2 = (W1 - F + 2P) / S + 1 \quad (3.1)$$

$$\text{height} : H2 = \frac{H1 - F + 2P}{S + 1} \quad (3.2)$$

$$\text{depth} : D2 = K \quad (3.3)$$

Cette structure présente de nombreux avantages lors du traitement de grandes données d'entrée telles que les images. Comme un filtre relativement petit peut être utilisé sur l'ensemble de l'image, le nombre de paramètres diminue fortement. Cependant, en utilisant de nombreux filtres sur l'image, différents filtres peuvent être entraînés à rechercher différentes caractéristiques. De plus, en utilisant des filtres de tailles différentes dans différentes couches, on peut facilement construire un modèle qui détecte des caractéristiques à différentes échelles dans l'image. [4]. Une fois qu'une couche convolutive a traité et restructuré les données de la couche d'entrée ou de la couche précédente, le volume de sortie est souvent envoyé à travers une

couche ReLU (Rectified-Linear). Elle applique une fonction d'activation par élément, le plus souvent une fonction $\max(0, x)$ qui fixe toutes les valeurs négatives à zéro. Cela devient la fonction d'activation de la couche qui introduit la non-linéarité, ce qui est crucial pour la capacité du réseau à mettre en correspondance différents types de fonctions. Il est important de noter que cette couche laisse la taille de la sortie intacte. [4]. Afin de réduire progressivement la taille spatiale du volume de sortie et de diminuer le nombre de paramètres à calculer à l'intérieur du réseau, il est courant d'insérer périodiquement ce que l'on appelle une couche de mise en commun entre les couches de convolution successives. L'opération de réduction de l'échantillonnage effectuée par la couche est également importante pour empêcher l'adaptation excessive, lorsque le réseau s'habitue trop aux images sur lesquelles il est entraîné et obtient de moins bons résultats sur d'autres images. La couche redimensionne les données dans l'espace en balayant chaque tranche de profondeur dans le volume d'entrée et en appliquant une opération maximale. Tout comme la couche convolutive, la couche de mise en commun possède un noyau d'étendue spatiale de taille F , où l'opération locale maximale est appliquée, et un paramètre de foulée S . Il est important de noter que lorsque les noyaux balayent les différentes tranches de profondeur, la profondeur totale du volume reste inchangée. Si un volume de largeur $W1$, de hauteur $H1$ et de profondeur $D1$ est introduit dans une couche de mise en commun, la sortie produite a des dimensions :

$$\text{width} : W2 = (W1 - F) / S + 1 \quad (3.4)$$

$$\text{height} : H2 = (H1 - F) / S + 1 \quad (3.5)$$

$$\text{depth} : D2 = D1 \quad (3.6)$$

Cela change si l'on introduit un rembourrage nul, mais ce n'est pas courant pour les couches de piscines. des couches de mise en commun. D'autres opérations que le max peuvent également être utilisées, comme la moyenne ou la norme L2 moins. Cependant, il a été démontré que le max est le plus efficace dans la plupart des situations. situations. La fonction appliquée par cette couche étant fixe, elle n'introduit pas de nouveaux paramètres dans le modèle [17]. nouveaux paramètres au modèle [4].

Le réseau neuronal convolutif est souvent terminé par des couches de feed forward régulières, qui ont récemment adopté le nom de couches entièrement connectées. Comme nous l'avons mentionné, ces couches sont coûteuses à former car tous les neurones d'une couche sont connectés à tous les neurones de la couche précédente. Étant à la fin du réseau, l'entrée est généralement gérable et elles sont utilisées pour calculer la sortie finale. Cela signifie, par exemple, que les scores de classe pour une image classifier [4].

3.5.1 Composant du CNN :

Couche d'entrée :

Définit les spécifications de l'image d'entrée : largeur, hauteur, nombre de canaux de couleur (généralement 1 ou 3).

Convolution/filtering layer :

Crée une carte de caractéristiques pour prédire les probabilités de classe pour chaque caractéristique en appliquant un filtre qui analyse/balaie l'image entière, un lot (quelques pixels) à la fois. Dans une couche de convolution, un produit de convolution est appliqué, qui consiste simplement à multiplier chaque valeur de pixel de l'image par la valeur de pixel correspondante du filtre, puis à calculer la somme totale du produit.[5].

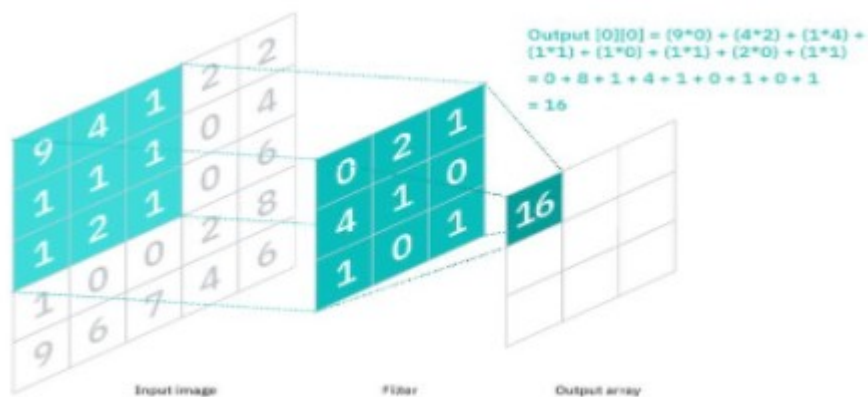


FIGURE 3.2 – Représentation schématique d'une couche de convolution dans un CNN.[5].

Couche de mise en commun :

Une couche de mise en commun fournit une opération typique de sous-échantillonnage qui réduit la dimensionnalité dans le plan des cartes de caractéristiques afin d'introduire une invariance de translation à de petits décalages et distorsions, et de diminuer le nombre de paramètres apprenables ultérieurs. Il est à noter qu'il n'y a aucun paramètre apprenable dans aucune des couches de mise en commun, alors que la taille du filtre, le pas et le remplissage sont des hyper paramètres dans les opérations de mise en commun, similaires aux opérations de convolution. Il est intéressant de mentionner qu'il existe 3 types différents de couches de mise en commun : Max, Mean et Sum pooling et la plus populaire des trois est la couche de mise en commun Max.

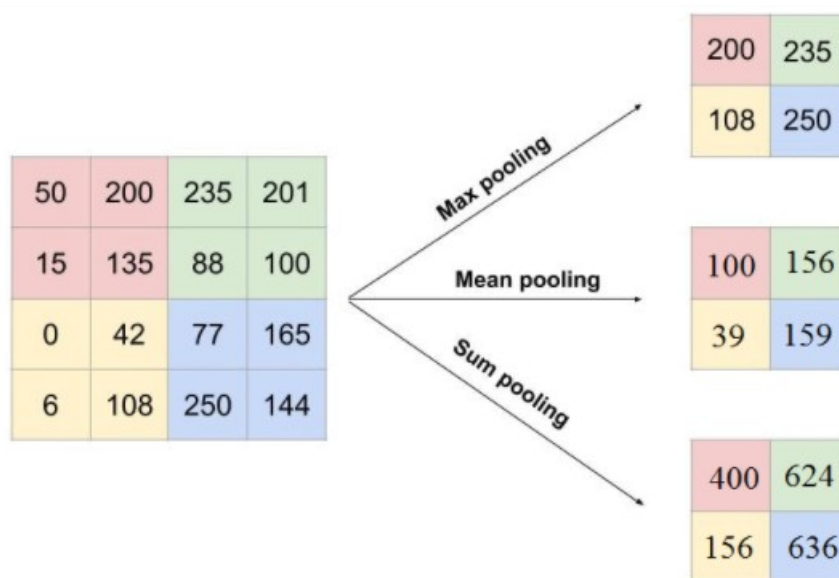


FIGURE 3.3 – Différence entre les types de pooling

Couche de correction linéaire (ReLU) :

Habituellement appelé ReLU (abréviation de Rectified Linear Unit) en référence à la fonction d'activation "unité de rectification linéaire", qui élimine toutes les valeurs négatives reçues en entrée et les remplace par des zéros selon la fonction suivante :

$$f(x) = \max(0, x)$$

Aplatissement :

consiste simplement à présenter les matrices de caractéristiques des images dans un vecteur, comme le montre la figure 3.4. Cela permettra la création d'une couche de neurones entièrement connectée. Chacune des valeurs de ce vecteur sera connectée aux neurones de la couche suivante du réseau.

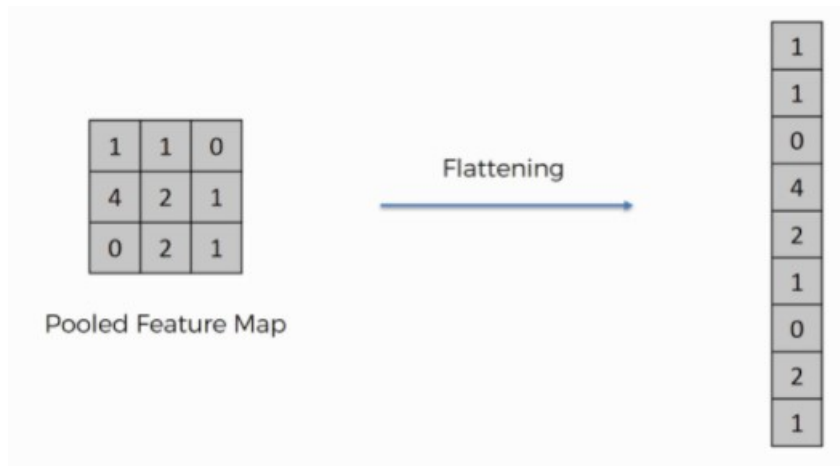


FIGURE 3.4 – the Flattening layer

Couche de sortie (Dense) :

Les cartes de caractéristiques de sortie de la couche finale de convolution ou de mise en commun sont généralement aplaties, c'est-à-dire transformées en un tableau unidimensionnel (1D) de nombres (ou vecteur), et connectées à une ou plusieurs couches entièrement connectées, également appelées couches denses, dans lesquelles chaque entrée est connectée à chaque sortie par un poids apprenable. Une fois que les caractéristiques extraites par les couches de convolution et échantillonnées par les couches de mise en commun sont créées, elles sont mises en correspondance par un sous-ensemble de couches entièrement connectées avec les sorties finales du réseau, telles que les probabilités de chaque classe dans les tâches de classification. La couche finale entièrement connectée possède généralement le même nombre de nœuds de sortie que le nombre de classes. Chaque couche entièrement connectée est suivie d'une fonction non linéaire, telle que ReLU, comme décrit ci-dessus. En arrière-plan, la couche dense effectue une multiplication matrice-vecteur. Les valeurs utilisées dans la matrice sont en fait des paramètres qui peuvent être formés et mis à jour à l'aide de la rétropropagation. La sortie générée par la couche dense est un vecteur de dimension 'm'. Ainsi, une couche dense est essentiellement utilisée pour modifier les dimensions du vecteur. Les couches denses appliquent également des opérations telles que la rotation, la mise à l'échelle et la translation sur le vecteur.

3.5.2 Quelques architectures CNN célèbres :

AlexNet :

En 2012, Alex Krizhevsky, Ilya Sutskever et Geoff Hinton ont remporté le concours de reconnaissance visuelle à grande échelle ImageNet avec une précision de 84,6 pour cent [6]. Le modèle a largement surpassé le deuxième finaliste (erreur de 16 dans le top 5, contre 26 pour le finaliste). Krizhevsky a utilisé des GPU pour former l'AlexNet, ce qui a permis une formation plus rapide des modèles CNN et a déclenché une vague d'intérêt et de nouveaux travaux basés sur les CNN.

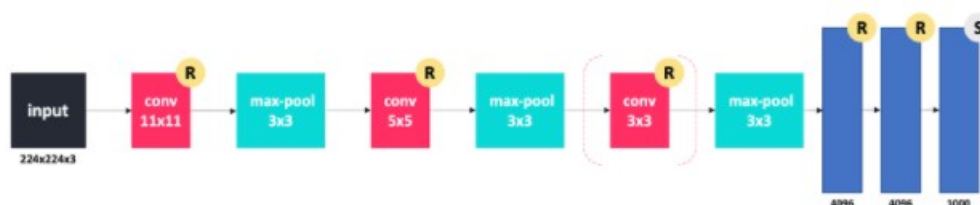


FIGURE 3.5 – Alex-net architecture

VGG16 :

Le modèle atteint une précision de 92,7 dans les cinq premiers tests d'ImageNet, qui est un ensemble de données de plus de 14 millions d'images appartenant à 1000 classes. Il a été proposé par Karen Simonyan et Andrew Zisserman du laboratoire du groupe de géométrie visuelle de l'université d'Oxford, dans le cadre de l'initiative de la Commission européenne. 2014[7].

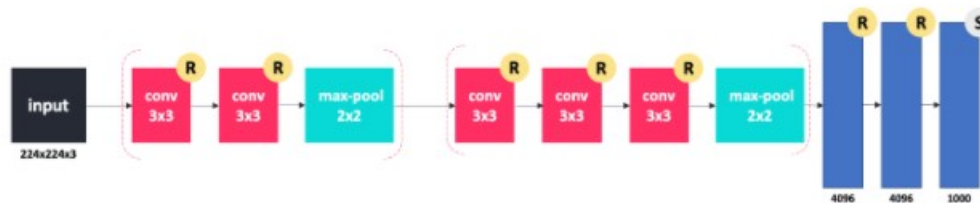


FIGURE 3.6 – Vgg16 architecture

VGG19 :

VGG-19 est un réseau neuronal convolutif de 19 couches de profondeur, capable de classer des images dans 1000 catégories d'objets, comme un clavier, une souris et de nombreux animaux. Le modèle a été entraîné sur plus d'un million d'images de la base de données Imagenet avec une précision de 92

ResNet :

Le réseau a été créé et pris connaissance par Microsoft. Avec une précision de 96,4 , ce modèle a remporté le concours ImageNet 2016. Il est bien connu en raison de sa profondeur (jusqu'à 152 couches) et de l'introduction de blocs résiduels.

3.6 Conclusion du chapitre

Dans ce chapitre, nous avons présenté un aperçu des concepts de base de l'apprentissage automatique et des réseaux de neurones artificiels, depuis la conception du perceptron jusqu'aux avancées actuelles. L'architecture globale d'un CNN et ses différentes couches et composants sont également présentés et décrits. couches et composants sont également présentés et décrits. Dans le chapitre suivant l'architecture proposée pour la tâche cible est présentée

Chapitre 4

Conception

4.1 Introduction au chapitre

Dans ce chapitre, nous allons parler en détail de l'architecture CNN utilisée et de la raison pour laquelle elle a été choisie parmi d'autres architectures.

4.2 Notre modele convolutif

Nous allons ici présenter notre propre modele convolutif et ceci pour montrer l'énorme apport que peut avoir l'utilisation de modeles pré-entraîné. Nous allons tout d'abord comparé trois architecture de convolution assez simple mais pour notre démonstration, suffiront largement. Nous prendrons ensuite ce modele sous l'appellation "ConvNet" et nous le comparerons a d'autres modeles pré-entraînés.

- 1. Un CNN avec une couche de convolution avec 32 filtres
- 2. Un CNN avec une couche de convolution avec 64 filtres
- 3. Un CNN avec une couche de convolution avec 128 filtres

Comme montré dans la figure 4.1 nous constatons que le modele avec 128 filtres est le plus performant pour notre cas d'études, car c'est le modele qui a été le plus performant dans l'ensemble de teste et l'ensemble de validation.

4.3 Comparaisons de différents modeles

Dans cette partie nous allons comparé differents CNN pré-entraîné et que nous avons transféré a notre cas d'étude.

Voici les CNN que nous allons comparés :

- InceptionV3
- MobileNet
- DenseNet
- VGG19

La figure 4.2 montre que mise a part notre modele ConvNet tous les autres modeles ont des resultats assez proches. Et cette écart est du a deux raisons principales :

- Le peu de données sur lequel notre modele a été entraîné. Les autres modeles ont déjà était entraîné sur des millions de données pour leur partie convolutives ce qui leurs a permis d'atteindre un haut niveau

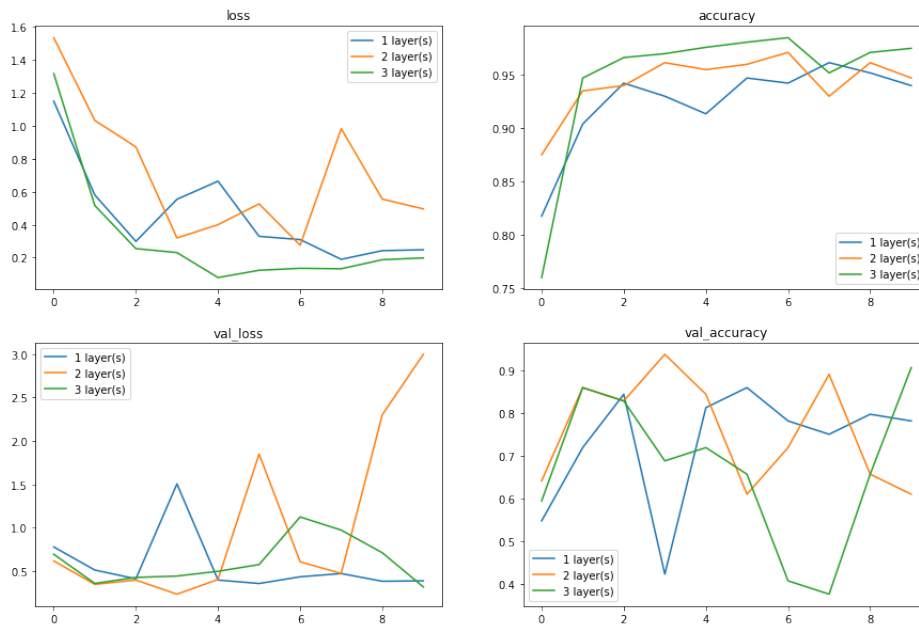


FIGURE 4.1

d'abstraction pour détecter des formes complexes, alors que notre modèle n'a été entraîné que sur les 938 images du jeu de données pour sa partie convolutive

- L'architecture de notre réseau est très simple comparée aux autres modèles. L'élaboration d'une architecture convolutive nécessite beaucoup de temps, de tests d'erreurs, et de rectification. Or avec notre approche dite "Transfer Learning" nous avons pu transférer un modèle pré-entraîné sur des données similaires à notre cas d'étude nous avons rapidement pu déployer des modèles performants tout en minimisant le temps et les ressources nécessaires à l'élaboration et l'entraînement d'un tel modèle.

Compte tenu des raisons citées plus haut et aussi le fait que notre application tourne en temps réels nous avons décidé d'opter pour l'utilisation de MobileNet. Il est à noter que MobileNet est de loin l'architecture la moins gourmande en ressources système parmi toutes les autres.

4.4 Pré-traitement :

La première étape vers la construction d'un modèle pour la tâche de suivi est le prétraitement des images. Le prétraitement de l'image fournit la visualisation nécessaire pour identifier et évaluer l'utilité de la procédure. Dans cette architecture, nous utilisons d'abord une cascade de Haar qui est une méthode d'apprentissage automatique pour la détection d'objets dans une image, largement utilisée dans le domaine de la vision par ordinateur. [8], Une cascade Haar entraînée, prenant une image en entrée, détermine si elle contient l'objet désiré, c'est-à-dire qu'elle effectue la tâche de classification, en divisant

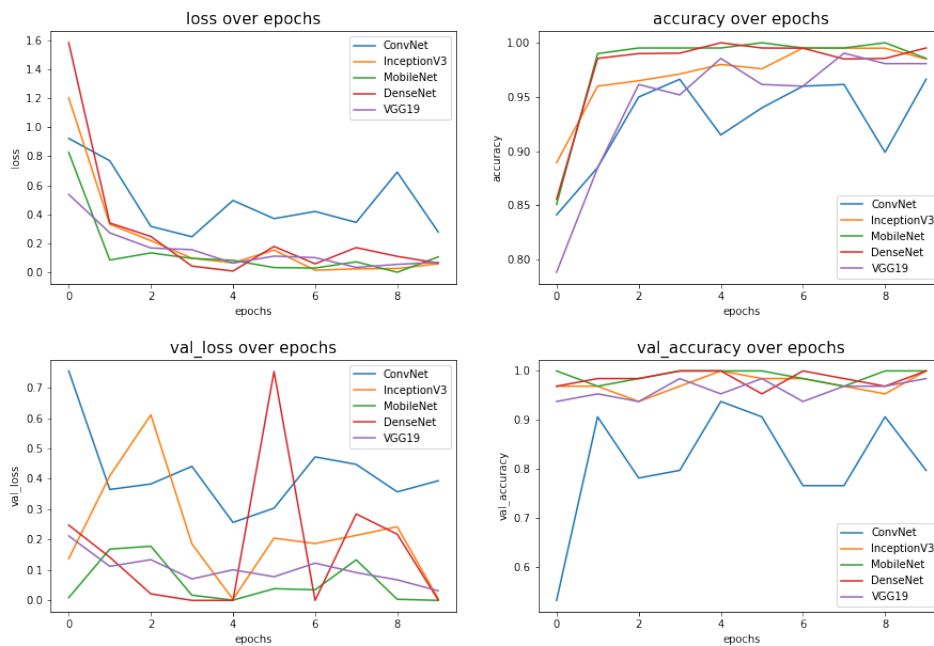


FIGURE 4.2 – Comparaison des performances des modeles

l'entrée en deux classes. Au premier stade, la fonction `dlib.get frontal face detector()` détermine le contour du visage. Ensuite, en utilisant la commande `dlib.shape predictor ("shape predictor 68 face landmarks.dat")`, nous définissons les caractéristiques du visage, puis nous utilisons un ROI OpenCV pour limiter la zone avec le visage dans le flux vidéo. See figure 4.3, qui montre les images de la vidéo avec l'œil limité par un rectangle de résolution.

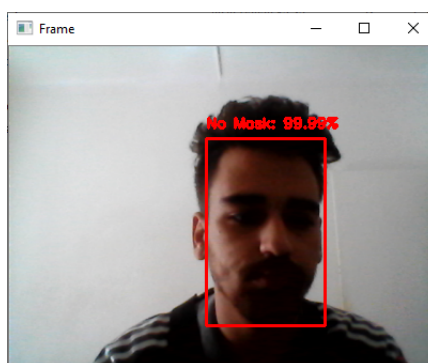


FIGURE 4.3 – extraction du visage

4.5 L'architecture CNN :

Compte tenu des résultats que nous avons conclu que le modèles mobileNet donne des résultats très proches des autres modèles mais l'avantage de mobileNet réside dans sa rapidité et sa légèreté : Il peut être déployé sur des systèmes relativement faible et tourné dans un environnement en temps réel. D'après le concepteurs de cette architecture : "MobileNetV2 est une architecture de réseau neuronal convolutif qui cherche à être performante sur les appareils mobiles. Il est basé sur une structure résiduelle inversée où les connexions résiduelles se trouvent entre les couches d'étranglement. La couche d'expansion intermédiaire utilise des convolutions légères en profondeur pour filtrer les caractéristiques comme source de non-linéarité. Dans l'ensemble, l'architecture de MobileNetV2 contient la couche initiale à convolution complète avec 32 filtres, suivie de 19 couches résiduelles à goulot d'étranglement.

Table 1. MobileNet Body Architecture

Type / Stride	Filter Shape	Input Size	
Conv / s2	$3 \times 3 \times 3 \times 32$	$224 \times 224 \times 3$	
Conv dw / s1	$3 \times 3 \times 32$ dw	$112 \times 112 \times 32$	
Conv / s1	$1 \times 1 \times 32 \times 64$	$112 \times 112 \times 32$	
Conv dw / s2	$3 \times 3 \times 64$ dw	$112 \times 112 \times 64$	
Conv / s1	$1 \times 1 \times 64 \times 128$	$56 \times 56 \times 64$	
Conv dw / s1	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$	
Conv / s1	$1 \times 1 \times 128 \times 128$	$56 \times 56 \times 128$	
Conv dw / s2	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$	
Conv / s1	$1 \times 1 \times 128 \times 256$	$28 \times 28 \times 128$	
Conv dw / s1	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$	
Conv / s1	$1 \times 1 \times 256 \times 256$	$28 \times 28 \times 256$	
Conv dw / s2	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$	
Conv / s1	$1 \times 1 \times 256 \times 512$	$14 \times 14 \times 256$	
$5 \times$	Conv dw / s1	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
	Conv / s1	$1 \times 1 \times 512 \times 512$	$14 \times 14 \times 512$
Conv dw / s2	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$	
Conv / s1	$1 \times 1 \times 512 \times 1024$	$7 \times 7 \times 512$	
Conv dw / s2	$3 \times 3 \times 1024$ dw	$7 \times 7 \times 1024$	
Conv / s1	$1 \times 1 \times 1024 \times 1024$	$7 \times 7 \times 1024$	
Avg Pool / s1	Pool 7×7	$7 \times 7 \times 1024$	
FC / s1	1024×1000	$1 \times 1 \times 1024$	
Softmax / s1	Classifier	$1 \times 1 \times 1000$	

FIGURE 4.4 – l’architecture mobile net

Voici un schéma pour simplifier les choses :

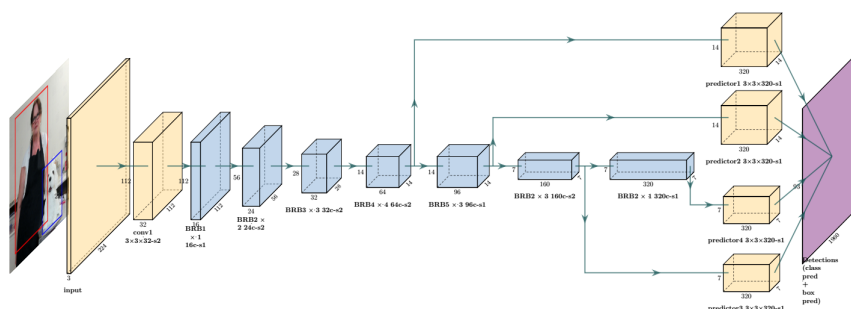


FIGURE 4.5 – Diagramme de l’architecture

4.6 Conclusion du chapitre

Dans ce chapitre, nous nous sommes concentrés sur la conception de la méthode proposée, nous avons d'abord présenté la phase de prétraitement qui comprend l'extraction de la région d'intérêt de l'image, puis nous avons plongé dans l'architecture du CNN proposé, dans le chapitre suivant, nous présenterons les différentes ressources que nous avons utilisées pour mettre en œuvre cette méthode.

Chapitre 5

Jeu de données et outils utilisés

5.1 Introduction au chapitre

Dans ce chapitre, nous allons parler des outils et du jeu de données utilisés pour la réalisation de ce projet.

5.2 Data Set : with-without-mask

Les masques jouent un rôle crucial dans la protection de la santé des individus contre les maladies respiratoires, car c'est l'une des rares précautions disponibles pour le COVID-19 en l'absence d'immunisation. Avec ce jeu de données, il est possible de créer un modèle permettant de détecter les personnes portant des masques, ne les portant pas, ou portant des masques de manière inappropriée (ce qui sera classer comme ne portant pas de masque). Ce jeu de données contient 938 images appartenant aux 2 classes, ainsi que leurs boîtes englobantes au format d'image png. Les classes sont :

- Avec masque
- sans masque



FIGURE 5.1 – échantillons des données

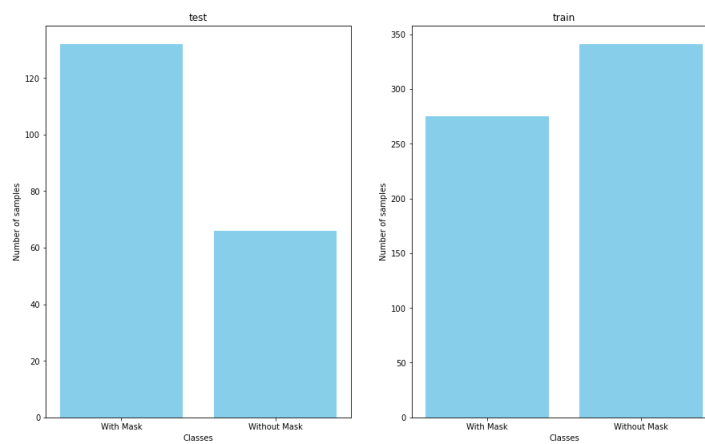


FIGURE 5.2 – Distribution des données

5.3 Outils et environnements utilisés :

5.3.1 Google Collaboratory :

Google Colab ou "le Colaboratoire" est un service gratuit en nuage hébergé par Google. pour encourager la recherche sur l'apprentissage automatique et l'intelligence artificielle, pour lesquels où l'obstacle à l'apprentissage et à la réussite est souvent l'exigence d'une énorme puissance de calcul. Il présente de nombreux avantages, notamment :

- Support de Python 2.7 et Python 3.6
- Accélération GPU gratuite
- Bibliothèques préinstallées : Toutes les principales bibliothèques Python comme TensorFlow, Scikit-learn, Matplotlib et bien d'autres encore sont préinstallées et prêtes à être importées.
- Construit au dessus de Jupyter Notebook
- Fonctionnalité de collaboration (fonctionne avec une équipe tout comme Google Docs) : Google Colab permet aux développeurs d'utiliser et de partager des carnets Jupyter entre eux sans avoir à télécharger, installer ou exécuter quoi que ce soit d'autre. entre eux sans avoir à télécharger, installer ou exécuter quoi que ce soit d'autre que qu'un navigateur.
- Prise en charge des commandes bash

5.3.2 OpenCV :

Open Source Computer Vision Library : Comme défini dans [25], OpenCV implémente une variété d'outils pour l'interprétation des images. Il est compatible avec l'Intel Image Processing Library (IPL), qui met en œuvre des opérations de bas niveau sur les images numériques. Malgré des primitives telles que la binarisation, le filtrage, les statistiques statistiques d'image, les pyramides, OpenCV est principalement une bibliothèque de haut niveau implémentant des algorithmes pour les techniques d'étalonnage (étalonnage de la caméra), la détection des caractéristiques (Feature) et le suivi (Optical). (Feature) et le suivi (Optical Flow), l'analyse de forme (Geometry, Contour des contours), l'analyse du mouvement (modèles de mouvement, estimateurs), la reconstruction 3D (morphing de la vue), l'analyse de l'image et l'analyse du mouvement. 3D (Morphing de vue), segmentation et reconnaissance d'objets (Histogramme, Modèles de Markov cachés Em- modèles de Markov cachés, objets propres). La caractéristique essentielle de cette bibliothèque, en plus de la fonctionnalité et de la qualité, est la performance. Les algorithmes sont basés sur des structures de données très flexibles (Dynamic Data Structures). Le logiciel logiciel fournit un ensemble de fonctions de traitement d'images, ainsi que des fonctions d'analyse d'images et de motifs. fonctions d'analyse d'images et de motifs. Les fonctions sont optimisées pour les processeurs d'architecture Intel Intel.

5.3.3 Keras :

Keras Keras est une bibliothèque logicielle open-source qui fournit une interface Python pour les réseaux de neurones artificiels. Keras sert d'interface à la bibliothèque TensorFlow. Jusqu'à la version 2.3, Keras prenait en charge plusieurs backends, notamment TensorFlow, Microsoft Cognitive Toolkit, Theano et PlaidML. 2.4, seul TensorFlow est pris en charge. Conçu pour permettre une expérimentation rapide avec des réseaux neuronaux profonds, il se concentre sur la convivialité, la modularité et l'extensibilité. Il a été développé dans le cadre de l'effort de recherche du projet ONEIROS (Open-ended Neuro-Electronic Intelligent Robot Operating System). principal auteur et mainteneur est François Chollet, un ingénieur de Google. Chollet est également l'auteur du modèle de réseau neuronal profond EXception [26]. [9].

5.3.4 TensorFlow :

TensorFlow est une plateforme open source de bout en bout pour l'apprentissage automatique. Elle possède un écosystème complet et flexible d'outils, de bibliothèques et de ressources communauté qui permet aux chercheurs de pousser l'état de l'art dans l'apprentissage automatique et aux développeurs de construire et de déployer facilement des applications alimentées par l'apprentissage automatique. et aux développeurs de créer et de déployer facilement des applications d'apprentissage automatique. Dans ce projet, il a été le principal outil pour construire et entraîner le modèle [10].

Chapitre 6

Discussion sur les résultats

6.1 introduction du chapitre

dans cette section de la thèse, nous allons discuter de la performance du modèle .

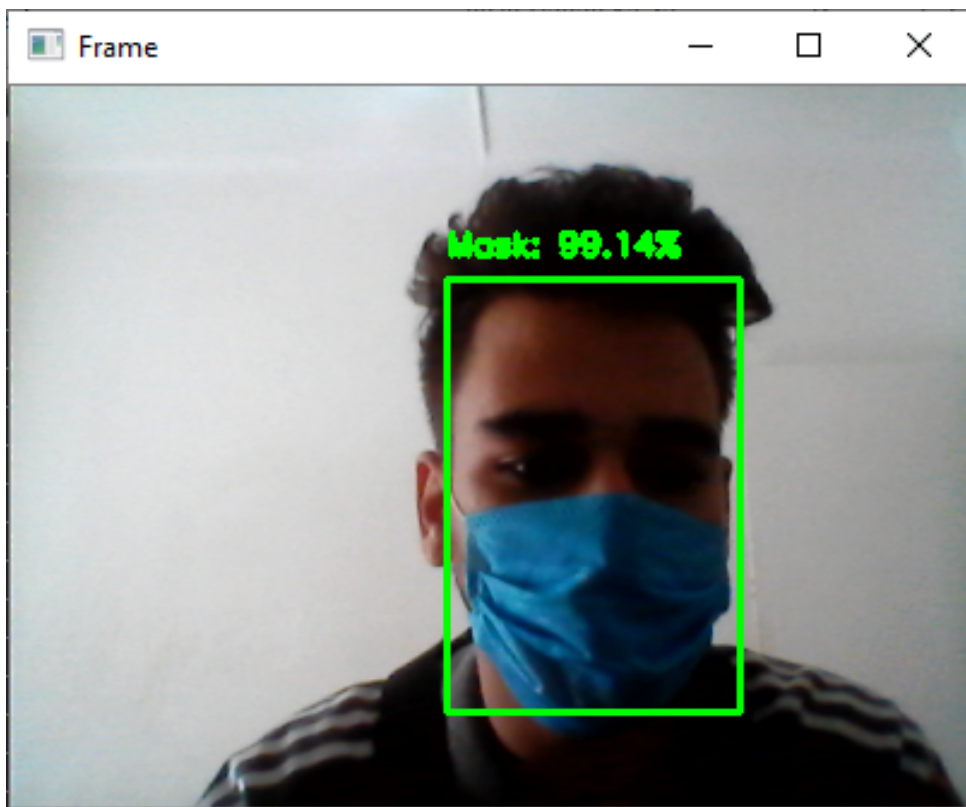


FIGURE 6.1 – Test du modèle

6.2 Evaluation du CNN

6.2.1 Perte (Loss)

La fonction de perte quantifie l'efficacité ou l'incapacité du prédicteur à classer les points de données d'entrée dans l'ensemble de données. Comme indiqué précédemment, plus la perte est faible, plus le classificateur réussit à modéliser la relation entre les données d'entrée et les étiquettes de classe de sortie (Rosebrock, 2017). [11]. La perte est cumulative par époque. Au début de chaque époque, la perte est nulle. Pour chaque calcul de la perte (donnée par la fonction de perte), la perte est ajoutée à la métrique de perte sur le graphique. Ce que l'on observe au fil du temps dans les résultats tracés en Figure 6.4, est la perte totale de la formation et de la validation qui diminue (généralement), ce qui signifie que les poids du réseau (dans ce cas, la tête de classificateur entièrement connectée sur le convnet) deviennent plus précis. Cependant, lorsque l'on lit une perte aussi faible, proche de zéro, il est évident que parce que les poids sont presque parfaitement adaptés aux données de formation et de validation, le modèle a franchi le point où il a été surajusté, ce qui conduit notre modèle à avoir des problèmes de généralisation.

Fonction de perte (Loss function)

Nous avons utilisé une fonction d'entropie croisée catégorielle éparsée puisque nos classes sont mutuellement exclusives (par exemple, lorsque chaque échantillon appartient exactement à une classe).

$$-\frac{1}{N} \sum_{i=1}^N \sum_{c=1}^C 1_{y_i \in C_c} \log P_{model}[y_i \in C_c] \quad (6.1)$$

Graphe de perte

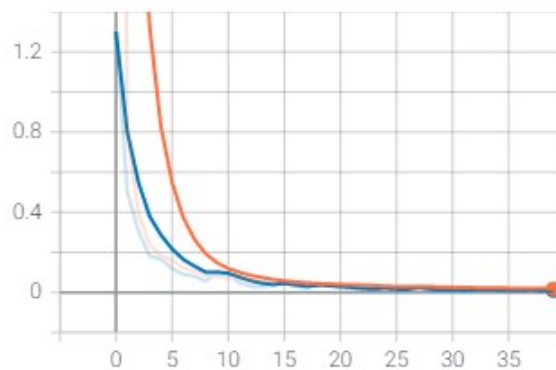


FIGURE 6.2 – Epoch Loss

6.2.2 Précision (Accuracy)

la précision de la formation augmente à chaque époque (figure 6.3), Cependant, un modèle qui donne de bons résultats sur des données d'apprentissage ne donne pas nécessairement de bons résultats sur des données qu'il n'a jamais vues auparavant (généralisation). La précision de l'apprentissage atteint 98 celle de la validation 99

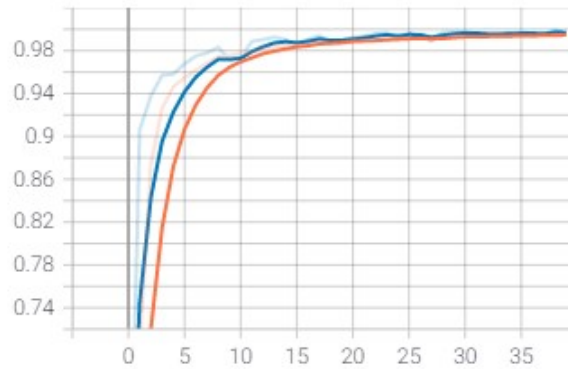


FIGURE 6.3 – Epoch Accuracy

6.2.3 Matrice de Confusion

Une matrice de confusion est un outil d'analyse prédictive. Plus précisément, il s'agit d'un tableau qui affiche et compare les valeurs réelles avec les valeurs prédites par le modèle. Dans le contexte de l'apprentissage automatique, une matrice de confusion est utilisée comme mesure pour analyser les performances d'un classificateur d'apprentissage automatique sur un ensemble de données. Une matrice de confusion permet de visualiser des paramètres tels que la précision, l'exactitude, la spécificité et le rappel.

n=165	Predicted: NO	Predicted: YES	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	

FIGURE 6.4 – Matrice de confusion

Conclusion

Ce travail a présenté une vue d'ensemble de 4 modèles pour la reconnaissance des masques médicaux, des algorithmes d'apprentissage profond. Nous avons choisi l'architecture MobileNet pour sa légèreté car nous avons pour objectif de réaliser une application de reconnaissance de masque en temps réel qui peut être déployée sur des systèmes relativement faibles : MobileNet peut même être déployé sur une raspberry Pi et tourné en temps réel. Nous avons proposé une pipeline comme suit : On recherche activement l'entrée vidéo pour trouver un visage avec l'architecture ResNet et à chaque fois qu'on trouve un visage on lance notre classifieur qui nous avons réalisé avec le Transfer Learning basé sur un modèle pré-entraîné sur la base ImageNet et que nous avons couplé à notre propre réseaux et nous l'avons entraîné sur la base Face-mask-detection dataset.

Notre travail est axé sur un sujet d'actualité : Notre application peut être déployée dans les endroits publics pour détecter les personnes ne portant pas un masque ce qui durant ces temps une atteinte à la santé publique et aide à la propagation de ce virus mortel. Nous sommes convaincus que notre application peut aider à combattre la propagation du COVID.

Bibliographie

- [1] Kevin P MURPHY. « A probabilistic perspective ». In : *Text book* (2012).
- [2] Li DENG et Dong YU. « Deep learning : methods and applications ». In : *Foundations and trends in signal processing* 7.3–4 (2014), p. 197-387.
- [3] Simon HAYKIN et N NETWORK. « A comprehensive foundation ». In : *Neural networks* 2.2004 (2004), p. 41.
- [4] Claus NEBAUER. « Evaluation of convolutional neural networks for visual recognition ». In : *IEEE transactions on neural networks* 9.4 (1998), p. 685-696.
- [5] Kazutaka UCHIDA, Masayuki TANAKA et Masatoshi OKUTOMI. « Coupled convolution layer for convolutional neural network ». In : *Neural Networks* 105 (2018), p. 197-205.
- [6] Alex KRIZHEVSKY, Ilya SUTSKEVER et Geoffrey E HINTON. « Imagenet classification with deep convolutional neural networks ». In : *Advances in neural information processing systems* 25 (2012), p. 1097-1105.
- [7] Christian SZEGEDY et al. « Going deeper with convolutions ». In : *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, p. 1-9.
- [8] P VIOLA et M JONES. « Face detection ». In : *IJCV* 57 (2004), p. 2.
- [9] François CHOLLET et al. *Keras*. <https://github.com/fchollet/keras>. 2015.
- [10] Martín ABADI et al. *TensorFlow : Large-Scale Machine Learning on Heterogeneous Systems*. Software available from tensorflow.org. 2015. URL : <https://www.tensorflow.org/>.
- [11] Orly Enrique APOLO-APOLO et al. « A cloud-based environment for generating yield estimation maps from apple orchards using UAV imagery and a deep learning technique ». In : *Frontiers in plant science* 11 (2020), p. 1086.