

UNIVERSITÉ DU QUÉBEC

MÉMOIRE PRÉSENTÉ À L'UNIVERSITÉ DU QUÉBEC À TROIS-RIVIÈRES

COMME EXIGENCE PARTIELLE DE
LA MAÎTRISE EN MATHÉMATIQUES ET INFORMATIQUE APPLIQUÉES

PAR

KOUIDER KARIM

DÉTECTION ET IDENTIFICATION DE VISAGES MASQUÉS

SEPTEMBRE 2024

Université du Québec à Trois-Rivières

Service de la bibliothèque

Avertissement

L'auteur de ce mémoire, de cette thèse ou de cet essai a autorisé l'Université du Québec à Trois-Rivières à diffuser, à des fins non lucratives, une copie de son mémoire, de sa thèse ou de son essai.

Cette diffusion n'entraîne pas une renonciation de la part de l'auteur à ses droits de propriété intellectuelle, incluant le droit d'auteur, sur ce mémoire, cette thèse ou cet essai. Notamment, la reproduction ou la publication de la totalité ou d'une partie importante de ce mémoire, de cette thèse et de son essai requiert son autorisation.

RÉSUMÉ

Dans le sillage de la pandémie mondiale de COVID-19, le port universel de masques a mis en lumière une problématique majeure pour les systèmes de reconnaissance faciale traditionnels, une problématique préexistante dans le domaine de la sécurité, où l'utilisation malicieuse de masques pose un défi. Notre mémoire vise à résoudre ce défi en se basant sur plusieurs expériences en utilisant une combinaison de techniques, notamment entre les différents types de réseaux de neurones convolutifs (CNN) et les techniques de détection de visages, avec une application d'expérimentations menant à notre objectif, qui est dans un premier lieu, la vérification et détection du port du masque en temps réel. Ce point est crucial, étant donné que dans le domaine de santé le port incorrect du masque peut nuire à son efficacité protectrice, et dans le domaine de sécurité, cette détection est primordiale pour la performance des systèmes de surveillance et de reconnaissance.

Les CNN, connus pour leurs performances en analyse d'images, sont utilisés pour l'extraction et l'encodage des caractéristiques des points du visage, tandis que le Haar-Cascade Classifier est employé en détection des visages dans les images des bases de données bien reconnue dans ce contexte de recherche, ces informations sont alors analysées et représentées sous forme d'histogrammes pour permettre l'identification de l'individu, que ce soit avec ou sans masque.

Ce travail de recherche présente une contribution notable au champ de la reconnaissance faciale tout en répondant à un défi contemporain majeur lié à la santé et la sécurité. Il offre une solution innovante à un problème nouvellement émergent, améliorant ainsi l'efficacité de la reconnaissance faciale en présence de masques.

Mots-clés : Reconnaissance faciale, Réseaux de neurones convolutifs (CNN), Masques, COVID-19, Apprentissage profond, Technique de Haar Cascade Classifier.

ABSTRACT

In the wake of the global COVID-19 pandemic, the universal wearing of masks has highlighted a significant issue for traditional facial recognition systems, an issue that has long existed in the field of security, where the malicious use of masks poses a challenge. This thesis aims to address this challenge by drawing on several experiments utilizing a combination of techniques, notably among different types of Convolutional Neural Networks (CNNs) and face detection techniques, with an application of experiments leading to our objective, which is primarily the verification and detection of mask wearing in real-time. This point is crucial, given that in the healthcare domain, incorrect mask wearing can compromise its protective effectiveness, and in the security domain, this detection is paramount for the performance of surveillance and recognition systems.

Convolutional Neural Networks (CNNs), renowned for their image analysis capabilities, are utilized for extracting and encoding facial feature points, while the Haar-Cascade Classifier is employed for facial detection in images from well-established databases in this research context. Subsequently, this information is analyzed and represented as histograms to enable individual identification, whether with or without a mask.

This research work presents a significant contribution to the field of facial recognition while addressing a major contemporary challenge related to health and security. It provides an innovative solution to a newly emerging problem, thereby enhancing the effectiveness of facial recognition in the presence of masks.

Keywords : Facial Recognition, Convolutional Neural Networks (CNN), Masks, COVID-19, Deep Learning, Image Processing, Haar Cascade Classifier Technique.

REMERCIEMENTS

Je tiens à exprimer ma profonde gratitude à toutes les personnes qui ont contribué à la réalisation de ce travail.

Tout d'abord, je voudrais remercier mon directeur de recherche, Mr. Ismail Biskri, pour son soutien inébranlable tout au long de ce processus. Sa patience, son expertise et ses précieux conseils ont été un guide essentiel pour moi. Son encouragement constant m'a permis de surmonter de nombreux défis au cours de cette recherche.

Je tiens également à exprimer ma gratitude envers ma mère, Mme Aidaoui Faiza, qui a grandement contribué au financement de cette recherche. Sa confiance en moi et son soutien m'ont permis de poursuivre cette étude malgré les obstacles. Son amour et son soutien ont été une source constante d'inspiration.

À ma femme, Asma Ouamane, je tiens à dire merci pour son amour, sa patience et son soutien indéfectible. Elle a été mon roc pendant cette période stressante, m'offrant encouragement et soutien émotionnel, et je lui en suis infiniment reconnaissant.

Je remercie également mes amis, Yacine Sari, Aimen Braham, Mouad Krim, Mohamed ElAmine Khoudour, Lakhdar Boulakrouche et Yacine Messouaf pour leur amitié, leur camaraderie et leurs conseils précieux. Leur présence et leur soutien ont été une source de réconfort et de motivation.

Enfin, je tiens à remercier tous ceux qui ont contribué de près ou de loin à ce travail, votre soutien a rendu cette recherche possible et je vous en suis profondément reconnaissant.

TABLE DES MATIÈRES

RÉSUMÉ	ii
ABSTRACT	iii
REMERCIEMENT	iv
LISTE DES TABLEAUX	1
LISTE DES FIGURES	2
PREMIER CHAPITRE - INTRODUCTION GÉNÉRAL	4
DEUXIÈME CHAPITRE - CONTEXTE DE RECHERCHE ET PROBLÉMATIQUE	7
2.1 INTRODUCTION À L'INTELLIGENCE ARTIFICIELLE	7
2.1.1 Introduction	7
2.1.2 Deep Learning	14
2.1.3 Big Data	22
2.1.4 Reconnaissance faciale	29
2.1.5 Impact du port de masques sur les systèmes de reconnaissance faciale	35
2.2 PROBLÉMATIQUE ET JUSTIFICATION DE LA RECHERCHE	36
2.3 OBJECTIFS ET DÉFIS DE RECHERCHE	36
2.4 APERÇU DE LA MÉTHODOLOGIE	37
2.5 CONCLUSION	38
TROISIÈME CHAPITRE - ÉTAT DE L'ART	39
3.1 INTRODUCTION	39
3.2 RÉSEAU DE NEURONES CONVOLUTIFS	40
3.3 TECHNIQUES DE DÉTECTION DE VISAGES	44
3.4 MÉTHODES DE DÉTECTION DE MASQUES	53
3.5 MODELS LES PLUS UTILISÉS EN RECONNAISSANCE FACIALE	59
3.6 CONCLUSION	65
QUATRIÈME CHAPITRE - MÉTHODOLOGIE	66
4.1 INTRODUCTION	66
4.2 ENVIRONNEMENT DE TRAVAIL	66

4.3	PRÉPARATION DE DONNÉES	68
4.3.1	Base de données pour la détection de visages masqués	69
4.3.2	Base de données pour la reconnaissance de visages masqués	72
4.4	DÉTECTION DE VISAGES	74
4.5	EXTRACTION DES ASPECTS DISTINCTIFS.....	77
4.6	CLASSIFICATION DES VISAGES	82
4.7	IDENTIFICATION	87
4.8	ARCHITECTURE DE NOTRE SYSTÈME	91
4.8.1	Architecture du système de détection de masques.....	91
4.8.2	Architecture du système de reconnaissance faciale	93
4.8.3	Avantages, optimisation et limites du système.....	95
4.9	CONCLUSION	95
	CINQUIÈME CHAPITRE - EXPÉRIMENTATIONS ET RÉSULTATS	97
5.1	INTRODUCTION.....	97
5.2	MODELE D'APPRENTISSAGE DÉVELOPPÉE.....	97
5.2.1	Description détaillée du modèle	99
5.3	EXPÉRIMENTATIONS ET RÉSULTATS	105
5.3.1	Approche de détection de masque	105
5.3.2	Approches de Reconnaissance de visage masqués	112
5.4	CONCLUSION	125
	SIXIÈME CHAPITRE - CONCLUSION GÉNÉRALE.....	126
	RÉFÉRENCE.....	128

LISTE DES TABLEAUX

Tableau 1 : Tableau comparatif des types d'intelligence artificielle.....	11
Tableau 2 : Tableau comparatif des types de modèles de Deep Learning	20
Tableau 3 : Tableau comparatif des types de Big Data.....	25
Tableau 4 : Comparaison des méthodes de reconnaissance faciale	33
Tableau 5 : Comparaison des Architectures CNN	42
Tableau 6 : Comparaison des techniques de détection de visages	51
Tableau 7 : Comparaison entre la première et deuxièmes approches.....	58
Tableau 8 : Comparaison des modèles de reconnaissance faciale.....	64
Tableau 9 : Comparaison entre la première et deuxièmes approches.....	123

LISTE DES FIGURES

Figure 1 : Diagramme de flux du processus d'apprentissage automatique	13
Figure 2 : Réseaux de neurones convolutifs typique.....	18
Figure 3 : Réseau de neurones récurrent	19
Figure 4 : Illustration d'autoencodeur	20
Figure 5 : Point clés d'un visage en reconnaissance faciale	31
Figure 6 : Comparaison 2D vs 3D en reconnaissance faciale	33
Figure 7 : Représentation de filtrage successif de la méthode Haar cascade.....	47
Figure 8 : évolution des efforts de recherche en détection de masques.....	54
Figure 9 : Diagramme YOLO en détection d'image	57
Figure 10 : Diagramme Faster R-CNN en detection d'objet	58
Figure 11 : Composition de la base de données MaskedFace-Net.....	69
Figure 12 : Statistiques d'âge et nationalité de la base de données (Million Celebs)	72
Figure 13 : Modèle d'apprentissage utilisé...../.....	99
Figure 14 : Entraînement de notre Modèle pour la détection de masque	106
Figure 15 : Graphique de la perte d'entraînement et validation (masque).....	106
Figure 16 : Graphique de l'exactitude d'entraînement et validation (masque).....	108
Figure 17 : Test de détection de masque sur image dans notre application.....	110
Figure 18 : Test de détections de masque en temps réel.....	110

Figure 19 : Extraction de la partie supérieure du visage.....	115
Figure 20 : Graphique de la perte d'entraînement et validation (reconnaissance).....	115
Figure 21 : Graphique de précision d'entraînement et validation (reconnaissance)	116
Figure 22 : Test de la première approche d'identification en temps réel.....	118
Figure 23 : Ajout de masque et augmentation de base de données.....	120
Figure 24 : Evolution d'entraînement et les données de validation de la deuxième approche...	120
Figure 25 : Test de la première approche d'identification en temps réel.....	122

PREMIER CHAPITRE - INTRODUCTION GÉNÉRALE

La pandémie de COVID-19 a entraîné des changements importants dans notre société, dont le plus visible est probablement celui du port généralisé du masque. Elle a mis en évidence un élément crucial de la reconnaissance faciale : l'efficacité de la reconnaissance des visages partiellement masqués. Les systèmes de reconnaissance faciale ont historiquement été appliqués à des visages non obstrués. La détection et la reconnaissance des visages masqués ont toujours posé problème, tant d'un point de vue technique qu'en raison des préoccupations liées à la sécurité, à la santé publique et à la protection de la vie privée.

Notre recherche porte sur le développement et le test d'un système d'identification et de détection des visages masqués. L'objectif est de créer un système robuste et efficace qui fonctionne dans des conditions naturelles où le port du masque est devenu une routine. Ce travail de recherche s'inscrit dans un contexte où les systèmes de reconnaissance faciale sont utilisés dans divers secteurs, tels que la sécurité publique, l'authentification personnelle, et même les soins de santé, afin de répondre aux exigences nouvelles imposées par les circonstances actuelles.

L'analyse de la littérature révèle que, jusqu'à récemment, la majorité des travaux sur la reconnaissance faciale se concentraient sur des visages entièrement visibles. Les systèmes traditionnels comme ceux basés sur l'algorithme de **Viola-Jones** ou des caractéristiques manuelles, bien qu'efficaces dans des environnements contrôlés, ne sont pas conçus pour gérer les conditions dynamiques imposées par le port du masque. Toutefois, avec l'avènement de nouvelles approches basées sur le **Deep Learning** et les **réseaux de neurones convolutifs (CNN)**, de récents travaux ont commencé à se pencher sur la détection des visages partiellement masqués. Les modèles comme **Faster R-CNN** et **YOLO** ont démontré leur efficacité dans la détection d'objets et de visages dans des images complexes, et ils représentent des avancées prometteuses dans ce domaine. Nous explorerons en profondeur ces méthodes dans la section dédiée à l'état de l'art [1].

Plus particulièrement, notre revue de la littérature mettra en lumière les technologies récentes qui ont transformé la reconnaissance faciale, notamment les réseaux de neurones convolutifs, l'apprentissage profond et l'utilisation de grandes bases de données d'images. Ces approches offrent de nouvelles perspectives pour surmonter les défis posés par les visages masqués. Par exemple, l'architecture **FaceNet**, développée par Google, permet d'extraire des représentations faciales robustes en utilisant des embeddings de visage, ce qui a révolutionné la manière dont les visages sont reconnus et classifiés, même lorsqu'ils sont partiellement cachés.

Notre système proposé repose sur deux axes principaux : la détection des visages dans les images via le classificateur **HaarCascade**, et l'extraction et l'encodage des caractéristiques faciales par des **réseaux neuronaux convolutifs pré-entraînés** et sur mesure. Nous avons également intégré des techniques d'**augmentation de données**, de **prétraitement d'images** et de **combinaison de modèles d'apprentissage** afin d'améliorer la robustesse et l'efficacité de notre système dans différents scénarios. Ces optimisations sont particulièrement importantes pour garantir que notre système puisse fonctionner efficacement même avec des images de faible qualité ou des angles de vue non conventionnels, ce qui est souvent le cas dans des environnements réels.

Nos expérimentations s'appuient sur deux bases de données majeures : « **MaskedFace-Net** » [2], une base de données contenant des images de visages masqués dans différents styles et contextes, et « **MillionCelebs** » [3], une vaste base de données contenant plus d'un million d'images de visages de célébrités. Ces bases de données sont essentielles pour entraîner et tester notre système, car elles couvrent une large gamme de conditions d'éclairage, d'angles et de types de masques, ce qui nous permet de tester la robustesse de notre approche dans des conditions réelles.

L'une des principales contributions de notre travail sera de démontrer comment les systèmes de reconnaissance faciale peuvent être adaptés pour faire face aux nouveaux défis posés par la pandémie. En adaptant les technologies existantes pour améliorer la reconnaissance des visages masqués, nous visons à renforcer la sécurité publique et à améliorer les systèmes de surveillance, tout en respectant les protocoles de santé publique en place.

Outre l'introduction, le mémoire se poursuit par une analyse approfondie du **contexte de recherche** et de la **problématique** associée, que nous examinerons dans le deuxième chapitre. Ce chapitre explorera également l'importance de la reconnaissance faciale dans divers domaines tels que la sécurité et la santé, ainsi que les défis techniques et éthiques auxquels ces systèmes sont confrontés. Ensuite, le **troisième chapitre** présentera un **état de l'art** détaillé des techniques actuelles de reconnaissance faciale, en insistant particulièrement sur les travaux récents sur la détection des visages masqués, et comment les nouvelles architectures de réseaux neuronaux et les algorithmes de détection d'objets, tels que **YOLO** et **SSD**, ont amélioré les performances des systèmes dans des environnements complexes.

Le **quatrième chapitre** décrira en détail la **méthodologie** employée pour concevoir, entraîner et tester notre système, en soulignant les choix techniques et les défis rencontrés lors du développement. Le **cinquième chapitre**, quant à lui, présentera les **résultats expérimentaux** obtenus, avec une analyse comparative des performances de notre système par rapport à d'autres systèmes existants. Cette analyse est essentielle pour démontrer la validité de notre approche, et les résultats seront discutés en termes de taux de réussite, de limitations et de scénarios d'utilisation réels.

Enfin, dans le dernier chapitre, nous proposerons une **synthèse générale** de notre travail, mettant en évidence les principales contributions de notre recherche et discutant des **perspectives futures** pour la reconnaissance faciale dans un monde post-pandémie. Nous suggérerons également des pistes pour des améliorations ultérieures, notamment en ce qui concerne l'intégration de nouvelles techniques d'apprentissage automatique et d'autres systèmes biométriques pour renforcer la fiabilité et la précision des systèmes de reconnaissance faciale dans des environnements complexes.

DEUXIÈME CHAPITRE - CONTEXTE DE RECHERCHE ET PROBLÉMATIQUE

2.1 INTRODUCTION À L'INTELLIGENCE ARTIFICIELLE

2.1.1 Introduction

L'intelligence artificielle (IA) est aujourd'hui omniprésente dans notre vie moderne et s'applique à un large éventail d'applications : médecine, fabrication, automobile et, plus récemment, sécurité et surveillance. En termes simples, l'IA décrit la capacité d'une machine à réaliser des performances cognitives extrêmement complexes qui exigent généralement une intelligence humaine, comme le raisonnement, la reconnaissance visuelle, la compréhension du langage naturel et la prise de décision. Les progrès de l'informatique, de la statistique et des neurosciences sont également importants pour le développement de l'IA, tout comme la capacité à concevoir des algorithmes capables de simuler la pensée humaine et les processus d'apprentissage. L'apprentissage automatique et l'apprentissage profond sont quelques-uns des sous-domaines sur lesquels s'appuie l'IA actuelle pour qu'une machine puisse apprendre à partir de grandes quantités de données. Les concepts d'IA remontent aux années 1950, lorsque les premiers algorithmes de résolution de problèmes ont été développés, ainsi que les systèmes experts pour effectuer des tâches spécialisées. Cependant, avec l'avènement de la puissance de calcul dans les années 2000 et 2010, l'IA a évolué dans des proportions jamais vues auparavant. Il en résulte des systèmes intelligents capables d'effectuer de nombreuses tâches avec beaucoup plus d'autonomie, notamment dans le domaine désormais très répandu de la reconnaissance faciale, de la sécurité publique à la personnalisation des services. Aujourd'hui encore, les applications de l'IA se multiplient, alimentant à la fois l'innovation technologique et les débats éthiques, sociaux et politiques. La reconnaissance faciale serait probablement l'une des technologies les plus prometteuses et les plus controversées à l'ère de l'IA et dans un contexte où la sécurité et la surveillance deviennent de plus en plus vitales.

Définition de l'intelligence artificielle :

L'intelligence artificielle peut donc être définie comme la capacité d'une machine à manifester une ressemblance avec l'intelligence humaine. Cela implique la capacité de reconnaissance de formes, la prise de décision, la compréhension à partir de données et l'adaptation à de nouvelles situations. Comme l'indiquent Stuart Russell et Peter Norvig dans leur ouvrage phare, *Artificial Intelligence : A Modern Approach*, l'IA peut être considérée comme « l'étude des agents qui reçoivent des perceptions de l'environnement et exécutent des actions » [4].

Exemples d'applications de l'IA :

- **Reconnaissance d'images** (par exemple, dans la reconnaissance faciale) [5].
- **Assistants vocaux** (comme Siri ou Alexa).
- **Voitures autonomes** (utilisant des algorithmes de décision basés sur des capteurs et des caméras) [4].
- **Systèmes de recommandation** (comme ceux utilisés par Netflix ou Amazon).

Types d'intelligence artificielle

L'intelligence artificielle peut être classée en trois grandes catégories en fonction de ses capacités : l'IA faible, l'IA forte, et l'IA superintelligente. Ces distinctions sont essentielles pour comprendre l'évolution des systèmes intelligents.

1. L'IA faible (ou IA étroite)

L'IA faible, autrement appelée IA étroite ou spécifique, représente les types d'IA que la plupart des applications actuelles utilisent. Elle est configurée avec un degré élevé de précision pour des tâches spécifiques, mais incapable de généraliser ou de s'adapter à d'autres tâches. Un exemple typique est un assistant virtuel comme Siri, qui est bon pour répondre aux questions, mais ne peut pas conduire de voiture de manière autonome ou rédiger des essais sans y être invité [4].

Caractéristiques de l'IA faible :

- Ne peut résoudre que des problèmes spécifiques pour lesquels elle a été programmée.
- Utilise des algorithmes d'apprentissage automatique pour s'améliorer dans une tâche donnée.
- N'a pas de conscience ni d'intelligence générale [6].

Exemples d'IA faible :

- Les assistants vocaux comme Google Assistant et Alexa, qui interprètent le langage naturel pour répondre à des questions [5].
- Les systèmes de recommandation d'Amazon ou Netflix, qui suggèrent des produits ou des films basés sur les préférences de l'utilisateur [4].

2. L'IA forte (ou IA générale)

L'IA forte, parfois appelée IA générale, est une forme d'intelligence artificielle capable d'imiter le raisonnement et la prise de décision humains dans toutes sortes de tâches, quelle que soit leur nature. À l'exception de l'IA faible, l'IA forte aurait la capacité de s'adapter à de nouvelles situations sans nécessiter de programmation spécifique.

Bien que l'IA forte soit une conception théorique, les chercheurs en intelligence artificielle y travaillent activement. Le défi consiste à doter une machine d'une intelligence générale qui lui permettrait d'effectuer n'importe quelle tâche cognitive qu'un humain peut effectuer [4]. L'IA forte pourrait probablement jouer à des jeux, résoudre des problèmes mathématiques complexes ou même comprendre et interpréter les émotions humaines.

Caractéristiques de l'IA forte :

- Capacité à généraliser l'apprentissage à de nouvelles tâches sans nécessiter de reprogrammation [6].
- Peut raisonner et résoudre des problèmes dans des environnements variés.
- Possède une certaine forme de conscience et d'intelligence générale [4].

Exemples théoriques :

- Une IA capable de participer à des conversations complexes sur divers sujets.
- Un robot capable de cuisiner en suivant une recette et d'apprendre de ses erreurs.

3. L'IA superintelligente

La superintelligence est une intelligence artificielle qui surpasse considérablement l'intelligence humaine générale dans tous les domaines. Ce type d'IA a souvent été décrit comme étant une écriture philosophique et éthique vers l'avenir de l'intelligence artificielle et généralement considéré comme une étape ultime dans le développement de l'IA. L'IA superintelligente sera capable de résoudre les problèmes complexes dont la conception même dépasse la capacité de création de l'esprit humain [7].

Caractéristiques de l'IA superintelligente :

- Intelligence exponentiellement supérieure à celle des humains dans tous les domaines, y compris la créativité, la résolution de problèmes, et la prise de décisions [7].
- Capable d'auto-amélioration rapide.
- Peut poser des risques existentiels selon certains chercheurs, car ses objectifs pourraient entrer en conflit avec ceux de l'humanité.

Exemples fictionnels :

- L'intelligence artificielle HAL 9000 dans *2001 : L'Odyssée de l'espace* [4].
- Les IA dans *Terminator*, où des machines superintelligentes surpassent les capacités humaines, (voir tableau 1).

Type d'IA	Capacités	Exemples actuels	Limites
IA faible (ou étroite)	Exécution de tâches spécifiques avec une grande précision	Siri, Alexa, Systèmes de recommandation	Incapable de généraliser au-delà des tâches programmées
IA forte (ou générale)	Capacités cognitives générales similaires à celles d'un humain	Hypothétique	Encore au stade théorique ; aucune implémentation réelle
IA superintelligente	Intelligence dépassant celle des humains dans tous les domaines	Fictionnelle	Risque potentiel pour l'humanité ; théorie pure

Tableau 1 : Tableau comparatif des types d'intelligence artificielle

Les sous-domaines de l'intelligence artificielle :

L'IA est un domaine vaste qui couvre plusieurs sous-disciplines spécialisées, chacune ayant ses propres applications et techniques. Voici quelques-uns des sous-domaines principaux de l'intelligence artificielle :

1. L'apprentissage automatique (Machine Learning)

L'apprentissage automatique est une sous-discipline clé de l'IA qui permet aux systèmes d'apprendre à partir des données sans être explicitement programmés pour chaque tâche. Selon [6], l'apprentissage automatique est devenu la méthode dominante pour résoudre de nombreux problèmes d'IA modernes.

Types d'apprentissage automatique :

- **Apprentissage supervisé** : Les modèles sont formés avec des ensembles de données étiquetés, et le modèle apprend à prédire la sortie correcte pour de nouvelles données [4], (voir figure 1).
- **Apprentissage non supervisé** : Ici, il n'y a pas d'étiquettes dans les données. Le modèle tente de découvrir des structures cachées, telles que des regroupements ou des associations entre les données [6].
- **Apprentissage par renforcement** : Un agent apprend à interagir avec un environnement et reçoit des récompenses ou des punitions en fonction des actions qu'il entreprend. Il apprend à maximiser sa récompense globale [8].

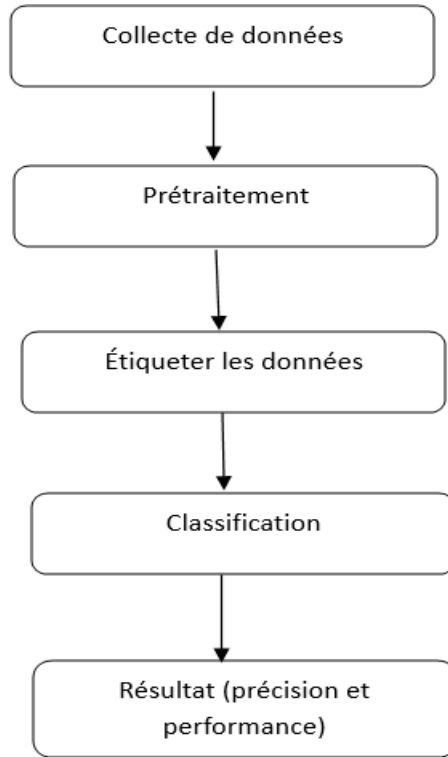


Figure 1 : Diagramme de flux du processus d'apprentissage automatique [39]

2. Le traitement du langage naturel (NLP)

Le traitement du langage naturel permet aux machines de comprendre, d'interpréter et de répondre au langage humain. Des applications telles que les traducteurs automatiques et les assistants personnels intelligents tirent parti des techniques du traitement du langage naturel pour traiter et comprendre les requêtes vocales [9].

3. La vision par ordinateur

La vision par ordinateur est la capacité des machines à comprendre des choses visuellement. Elle implique l'utilisation d'autres fonctionnalités telles que la reconnaissance d'objets, la détection de visages, la segmentation d'images et l'analyse vidéo. Les grandes avancées réalisées dans ce domaine sont associées à l'application de réseaux neuronaux convolutifs. [10].

4. Les systèmes experts

Les systèmes experts sont des programmes d'IA conçus pour imiter le raisonnement humain dans des domaines spécialisés tels que la médecine, la finance ou le droit. Ces systèmes utilisent des bases de connaissances et des règles d'inférence pour résoudre des problèmes complexes dans des environnements bien définis [4].

5. Les agents intelligents

Les agents intelligents sont des entités autonomes qui perçoivent leur environnement via des capteurs et agissent sur cet environnement pour atteindre leurs objectifs. Ces agents peuvent être utilisés dans des jeux vidéo, des simulations économiques ou des systèmes de gestion d'énergie [4].

2.1.2 Apprentissage profond (Deep Learning)

L'apprentissage profond est une classe d'apprentissage automatique qui repose sur l'utilisation de réseaux neuronaux artificiels profonds. Une interconnexion de plusieurs couches de ces systèmes permet au modèle d'apprendre des représentations de plus en plus abstraites des données à mesure qu'il passe à travers les couches du modèle. Le terme « profond » fait référence au nombre de couches intermédiaires entre la couche d'entrée et la couche de sortie au sein d'un réseau neuronal [6].

L'apprentissage profond a récemment pris de l'ampleur en raison de l'augmentation des puissances de calcul, des grands volumes de données disponibles et des algorithmes de formation avancés. Il a également transformé la reconnaissance d'images, la traduction automatique, le traitement du langage naturel et bien sûr la reconnaissance faciale [5].

Définition de l'apprentissage profond

L'apprentissage profond est l'une des méthodologies d'apprentissage automatique qui utilisent des modèles basés sur des réseaux de neurones artificiels pour modéliser des données complexes. Ces modèles permettent d'effectuer un apprentissage automatique de la représentation hiérarchique des données, en capturant des informations plus abstraites et plus pertinentes pour chaque couche par rapport à la tâche. La capacité d'extraction de

caractéristiques complexes par l'apprentissage profond rend cette technologie particulièrement efficace pour le traitement de données non structurées telles que des images, des textes ou des vidéos [5]. Les réseaux de neurones utilisés dans le Deep Learning peuvent être conçus soit comme une couche entièrement connectée, soit comme des réseaux de convolution, selon la nature des données à traiter. Le processus d'apprentissage est basé sur la rétropropagation des erreurs, une méthode permettant d'ajuster les poids des connexions entre les neurones en fonction de l'écart entre la sortie du modèle et la vérité terrain [6].

Mathématiquement, pour un réseau à une seule couche, la sortie y est donnée par la formule:

$$y = f(W \cdot x + b)$$

Où :

- x est le vecteur d'entrée,
- W est la matrice des poids,
- b est le biais,
- f est une fonction d'activation non linéaire, par exemple la fonction **sigmoïde** ou **ReLU** (Rectified Linear Unit).

La fonction **sigmoïde** est définie comme suit :

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

Tandis que la fonction **ReLU** est définie par :

$$ReLU(z) = \max(0, z)$$

Ces fonctions d'activation introduisent la non-linéarité dans les réseaux de neurones, ce qui leur permet de modéliser des relations complexes dans les données.

Rétropropagation de l'erreur

Le processus d'apprentissage dans les réseaux de neurones se base sur l'algorithme de rétropropagation de l'erreur (**backpropagation**). Cet algorithme calcule l'erreur de prédiction pour chaque exemple de données, puis utilise le gradient de cette erreur par rapport aux poids du réseau pour ajuster les poids afin de minimiser l'erreur globale.

La **fonction de coût** $J(\theta)$ (souvent appelée fonction de perte) mesure la différence entre la sortie prédite et la sortie réelle. Par exemple, pour une régression, une fonction de coût courante est la **fonction d'erreur quadratique moyenne** :

$$J(\theta) = \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2$$

Où :

- m est le nombre d'exemples,
- $h_{\theta}(x^{(i)})$ est la prédiction pour l'exemple i ,
- $y^{(i)}$ est la valeur réelle pour cet exemple.

Exemples d'applications du Deep Learning :

- **Reconnaissance d'images** (utilisée pour des tâches comme la reconnaissance faciale) [10].
- **Traitement du langage naturel** (comme dans la traduction automatique ou la génération de texte).
- **Jeux vidéo** (les algorithmes de Deep Learning sont utilisés pour des tâches comme le contrôle d'agents autonomes dans des environnements virtuels, par exemple dans les jeux de stratégie) [11].

Types de modèles en apprentissage profond (Deep Learning)

Il existe plusieurs types de modèles de Deep Learning, chacun adapté à des tâches spécifiques en fonction de la structure des données et des objectifs d'apprentissage. Les modèles les plus courants sont les réseaux de neurones convolutifs (CNN), les réseaux de neurones récurrents (RNN), et les autoencodeurs.

1. Réseaux de neurones convolutifs (Convolutional Neural Networks, CNN)

Les réseaux neuronaux convolutionnels sont spécifiquement conçus pour traiter des données structurées sous forme de grille, par exemple des images. Les CNN utilisent des filtres de convolution pour extraire les caractéristiques locales d'une image, telles que les contours, les textures ou les formes. Ceux-ci sont ensuite combinés pour identifier des motifs et des objets complexes au sein d'une image [12]. Un CNN typique se compose de plusieurs couches de convolution, de pooling et de couches entièrement connectées. Ce type de réseau a été largement utilisé pour des tâches telles que la classification d'images, la détection d'objets et la reconnaissance faciale [10]. (voir figure 2)

Exemples d'applications des CNN :

- Reconnaissance faciale (utilisée par des systèmes comme FaceNet pour identifier des individus à partir d'images de visages).
- Classification d'images dans des bases de données comme ImageNet [10].

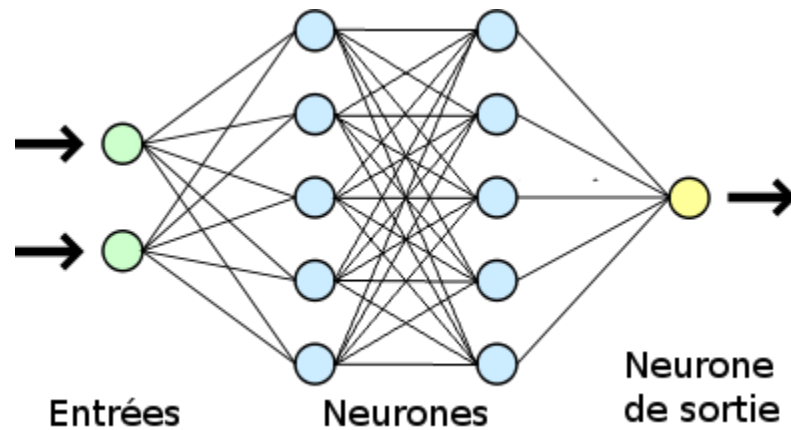


Figure 2 : Réseaux de neurones convolutifs typique [12].

2. Réseaux de neurones récurrents (Recurrent Neural Networks, RNN)

Les **réseaux de neurones récurrents (RNN)** sont conçus pour traiter des données séquentielles, telles que des séries temporelles ou des séquences de texte. Contrairement aux réseaux de neurones classiques, qui traitent chaque entrée indépendamment, les RNN conservent une mémoire des étapes précédentes dans une séquence, ce qui leur permet de tenir compte du contexte dans les prédictions. Cela les rend particulièrement adaptés aux tâches de traitement du langage naturel et à l'analyse de séquences temporelles [9]. (voir figure 3)

Les **long short-term memory (LSTM)** et les **gated recurrent units (GRU)** sont des variantes d'architectures RNN conçues pour mieux gérer les problèmes de longue dépendance et d'oubli dans les séquences [13].

Exemples d'applications des RNN :

- Traduction automatique (comme dans Google Translate).
- Génération de texte ou prédiction de séries temporelles [9].

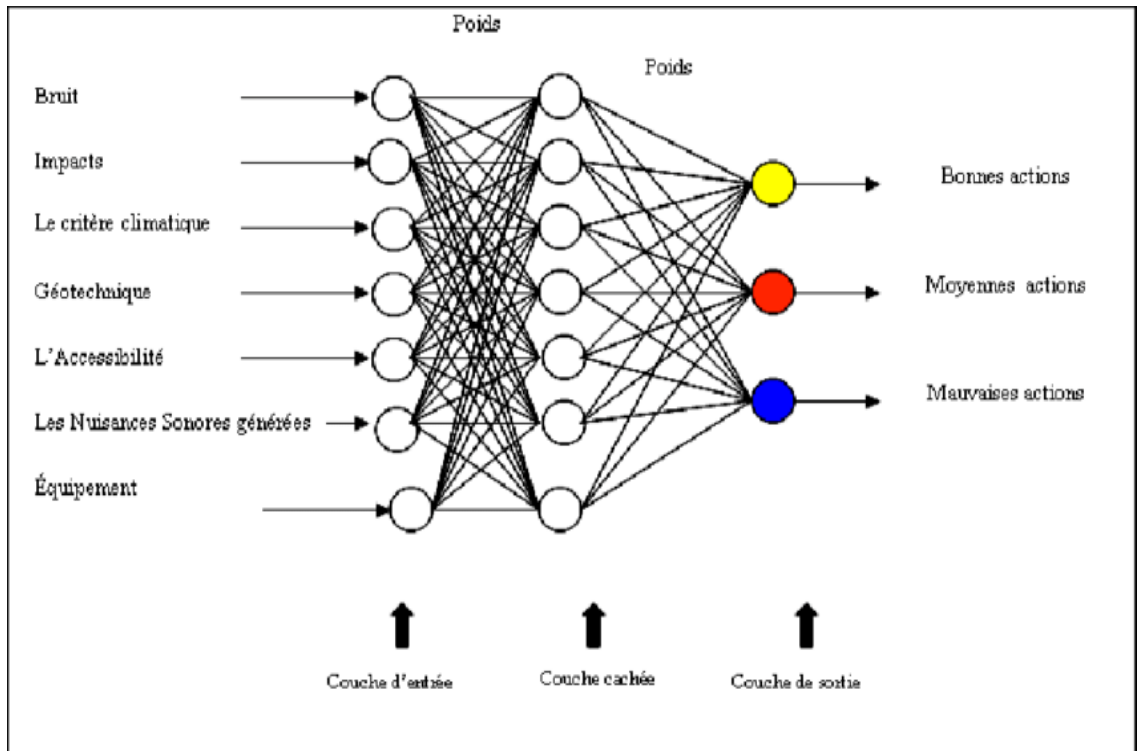


Figure 3 : Réseau de neurones récurrent [13].

3. Autoencodeurs

Les **autoencodeurs** sont une forme de réseaux neuronaux non supervisés qui sont utilisés pour apprendre la représentation des données compressées. Les modèles peuvent être divisés en deux parties : un encodeur, qui compresse les données d'entrée en une dimensionnalité inférieure ; et un décodeur qui tente de reconstruire les données d'origine à partir de cette représentation. Les applications les plus courantes des autoencodeurs sont liées à la réduction de la dimensionnalité, à la compression des données et à la détection des anomalies [14]. (voir figure 4)

Exemples d'applications des autoencodeurs :

- Compression d'images et réduction de dimensionnalité [14].
- Détection d'anomalies dans des systèmes industriels ou financiers.

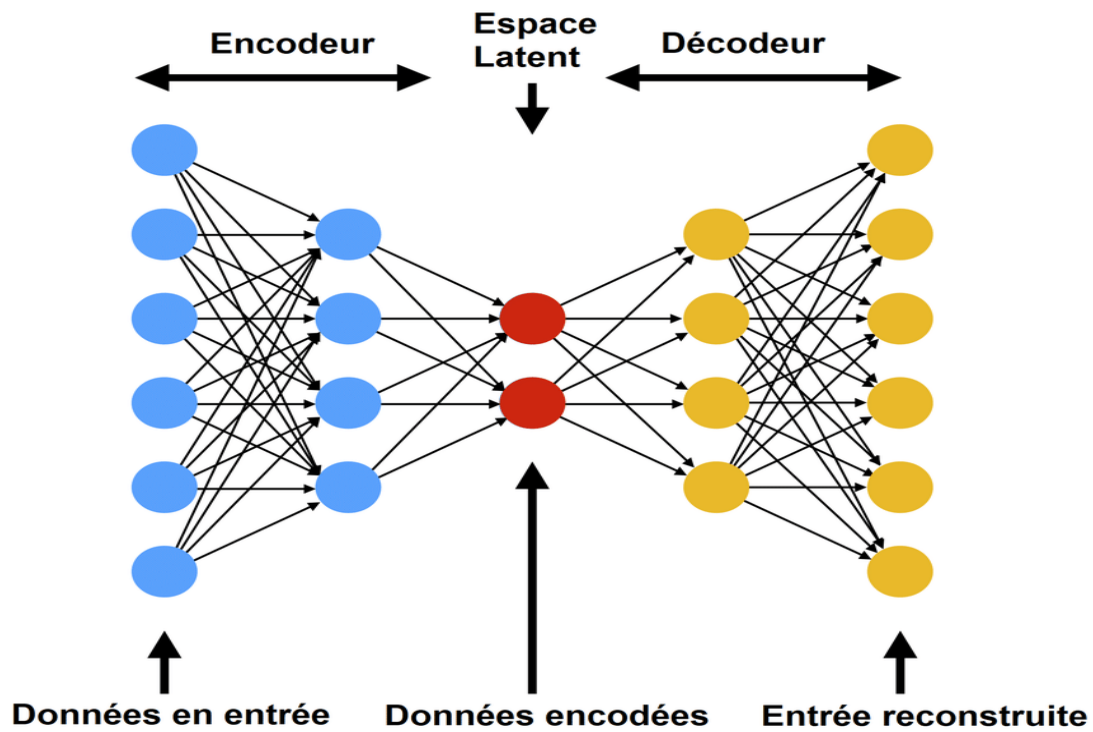


Figure 4 : Illustration d'autoencodeur [38]

Type de modèle	Caractéristiques principales	Applications principales	Avantages	Limites
CNN	Traite des données structurées (images, vidéos)	Reconnaissance d'images, détection d'objets	Très performant pour la vision par ordinateur	Difficile à appliquer aux données séquentielles
RNN (incl. LSTM et GRU)	Conserve la mémoire des séquences précédentes	Traitement du langage naturel, séries temporelles	Efficace pour les tâches séquentielles et les dépendances	Problème de vanishing gradient avec des séquences longues

Autoencodeurs	Apprend des représentations compressées des données	Compression de données, détection d'anomalies	Utile pour la réduction de dimensionnalité	Moins performant pour des tâches complexes de classification
----------------------	---	---	--	--

Tableau 2 : Tableau comparatif des types de modèles de Deep Learning

Avantages et limites du Deep Learning

Le Deep Learning a révolutionné l'intelligence artificielle en permettant des percées dans des domaines complexes comme la vision par ordinateur, le traitement du langage naturel, et les jeux [11]. Cependant, malgré ses succès, il présente également certaines limites.

Avantages du Deep Learning :

- **Performance supérieure :** Les modèles de Deep Learning surpassent souvent d'autres méthodes d'apprentissage automatique dans des tâches complexes comme la classification d'images ou la reconnaissance vocale [5].
- **Apprentissage non supervisé :** Les modèles comme les autoencodeurs peuvent apprendre des représentations sans nécessiter d'étiquettes explicites [14].
- **Capacité à traiter des données non structurées :** Contrairement aux méthodes traditionnelles, le Deep Learning peut traiter des données non structurées comme des images ou du texte sans nécessiter d'ingénierie manuelle des caractéristiques [6].

Limites du Deep Learning :

- **Dépendance aux données :** Les modèles de Deep Learning nécessitent de vastes ensembles de données pour être efficaces, ce qui peut poser des problèmes dans des contextes où les données sont rares [4].

- Consommation de ressources : Les réseaux de neurones profonds nécessitent des capacités de calcul importantes, notamment en termes de GPU et de mémoire [6].
- Manque d'explicabilité : Les modèles de Deep Learning sont souvent perçus comme des «boîtes noires», et il est difficile d'expliquer leurs décisions ou leurs prédictions [5].

2.1.3 Big Data

Le Big Data implique généralement des données volumineuses, diverses et à haut débit, d'une manière que les mécanismes traditionnels de gestion des données sont souvent incapables de traiter. Dans l'explosion actuelle des nouvelles technologies et de l'Internet des objets, les volumes de données produits à l'échelle mondiale ont atteint des échelles inégalées. Ces données peuvent provenir de sources telles que les réseaux sociaux, les transactions en ligne, les capteurs industriels, les appareils mobiles ou les caméras de surveillance. L'analyse des données doit permettre d'indiquer l'acquisition d'informations importantes qui permettront de prendre des décisions éclairées, plus particulièrement dans le domaine de la reconnaissance faciale, de l'intelligence artificielle et d'autres domaines de la recherche médicale [15].

Définition de Big Data

Le terme **Big Data** se définit souvent par les "5V" : **Volume**, **Vélocité**, **Variété**, **Véracité**, et **Valeur** [16].

- **Volume** : Quantité massive de données générées et stockées. Les entreprises peuvent désormais gérer des exaoctets (1 exaoctet = 10^{18} octets) de données.
- **Vélocité** : Vitesse à laquelle les données sont générées, souvent en temps réel. Par exemple, chaque minute, des millions de tweets, de messages et de transactions sont créés.
- **Variété** : Diversité des types de données. Les données peuvent être structurées (bases de données), semi-structurées (XML, JSON) ou non structurées (images, vidéos, texte libre).

- **Véracité** : Qualité des données, qui peut être affectée par des erreurs, du bruit ou des imprécisions.
- **Valeur** : L'importance des données réside dans les informations précieuses qu'elles peuvent fournir après analyse.

Exemples d'utilisation du Big Data :

- **Reconnaissance faciale** : Des systèmes comme ceux utilisés par les gouvernements ou les entreprises de sécurité utilisent de grandes quantités de données visuelles pour identifier des individus avec précision.
- **Analyse prédictive** : Utilisée dans des domaines comme la finance et la santé, l'analyse prédictive permet de tirer des conclusions à partir de vastes ensembles de données pour anticiper des événements futurs [15].
- **Personnalisation des services** : Les plateformes comme Netflix et Amazon utilisent le Big Data pour recommander des produits ou des films en analysant les habitudes des utilisateurs.

Les types de Big Data

Le Big Data peut-être classé en trois grandes catégories, selon la nature des données et leur mode de génération :

1. Données structurées

Les **données structurées** sont organisées dans des formats prédéfinis, généralement sous forme de tableaux ou de bases de données relationnelles. Elles sont facilement analysables à l'aide de requêtes SQL et autres outils de gestion de bases de données traditionnels. Chaque donnée a une place définie et respecte un schéma prédéfini, tel qu'une base de données clients avec des colonnes pour le nom, l'adresse et le numéro de téléphone.

Caractéristiques :

- Données bien organisées et structurées (par exemple, tableaux, fichiers CSV).
- Facilement analysables avec des méthodes traditionnelles.
- Volume souvent plus faible comparé aux données non structurées.

Exemples :

- Bases de données relationnelles (MySQL, PostgreSQL).
- Tableaux d'Excel.

2. Données semi-structurées

Les **données semi-structurées** sont des données qui ne suivent pas un schéma rigide comme les données structurées, mais qui contiennent des balises ou des métadonnées pour organiser les informations. Ces données peuvent inclure du texte libre ainsi que des informations balisées pour indiquer la structure.

Caractéristiques :

- Pas de structure rigide, mais des éléments de structure présents (balises XML, JSON).
- Utilisées dans des applications Web et mobiles.
- Requiert des outils spécifiques pour l'analyse.

Exemples :

- Fichiers XML, JSON.
- Données de logs d'applications.

3. Données non structurées

Les données non structurées constituent la plus grande composante du **Big Data**. Elles n'ont pas de structure prédéfinie et sont donc plus difficiles à analyser à l'aide de techniques traditionnelles. Cela inclut le contenu visuel tel que les images, les vidéos, le texte libre (pensez à tous les messages sur les réseaux sociaux) ou même les fichiers audios. Grâce aux progrès réalisés avec le **Deep Learning** et d'autres technologies d'intelligence artificielle, certains progrès ont été possibles en matière d'analyse de ces données non structurées.

Caractéristiques :

- Données très diverses (images, vidéos, texte).
- Nécessite des outils avancés, comme les réseaux de neurones, pour l'analyse.
- Représente environ 80 % du volume total des données générées [17].

Exemples :

- Publications sur les réseaux sociaux (Twitter, Facebook).
- Vidéos de caméras de surveillance.
- Images pour la reconnaissance faciale. (**Voir tableau 3**)

Type de données	Caractéristiques	Exemples	Méthodes d'analyse
Données structurées	Organisées dans un format prédéfini, facilement manipulables	Bases de données relationnelles, fichiers CSV	Requêtes SQL, outils traditionnels
Données semi-structurées	Partiellement organisées avec des balises ou métadonnées	XML, JSON, logs d'applications	Analyse XML, JSON parsers

Données non structurées	Non organisées, difficiles à analyser, volumineuses	Images, vidéos, texte libre	Analyse avec Deep Learning, Text Mining
--------------------------------	---	-----------------------------	---

Tableau 3 : Tableau comparatif des types de Big Data

Analyse du Big Data

L'analyse des Big Data est un moyen de décrire des informations utiles à partir d'ensembles de données volumineux et complexes. Certaines techniques incluent l'apprentissage automatique, le traitement du langage naturel et la reconnaissance d'images. Cela signifie à son tour de transformer ces informations en connaissances exploitables, permettant aux individus de prendre de meilleures décisions ou d'optimiser les processus commerciaux en conséquence.

L'analyse des données structurées peut être effectuée par le biais de requêtes **SQL** conventionnelles. Pour comprendre les ventes moyennes d'un produit particulier, dans une base de données, on pourrait utiliser une requête SQL simple :

$$\text{Moyenne des ventes} = \frac{\sum_{i=1}^n \text{ventes}_i}{n}$$

Pour les **données non structurées**, des techniques avancées comme le **Deep Learning** et les **algorithmes de clustering** sont nécessaires. Par exemple, dans le traitement des images, la **transformation de Fourier** est souvent utilisée pour analyser les fréquences spatiales des pixels dans une image, ce qui peut être formulé comme suit :

$$F(u, v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) \cdot e^{-j2\pi\left(\frac{ux}{M} + \frac{vy}{N}\right)}$$

Où :

- $f(x,y)$ est la valeur d'intensité du pixel à la position (x,y) ,
- $F(u,v)$ est la transformation de Fourier,
- M et N sont les dimensions de l'image.

Outils utilisés dans le Big Data

L'analyse du Big Data nécessite des outils spécialisés pour traiter, stocker et analyser ces données. Voici quelques-unes des technologies et plateformes couramment utilisées dans ce domaine :

1. Hadoop

Hadoop est un framework open source qui permet le traitement distribué de grandes quantités de données sur des clusters d'ordinateurs. Il repose sur un système de fichiers distribué appelé **HDFS** (Hadoop Distributed File System) et un modèle de programmation appelé **MapReduce**.

MapReduce permet de diviser une tâche d'analyse complexe en deux phases principales :

- **Map** : Divise les données en sous-ensembles plus petits et applique une fonction à chaque sous-ensemble.
- **Reduce** : Agrège les résultats des fonctions **Map** pour produire un résultat final.

2. Spark

Apache Spark est une autre plateforme open source qui offre une alternative plus rapide à Hadoop grâce à son traitement en mémoire (in-memory computing). Spark permet le traitement rapide des données, qu'il s'agisse de grandes quantités de données en lots ou de flux de données en temps réel.

3. NoSQL

Les bases de données **NoSQL** sont conçues pour gérer des données non structurées ou semi-structurées. Contrairement aux bases de données relationnelles, NoSQL n'impose pas de schéma fixe. Elles sont souvent utilisées dans des contextes de Big Data où la flexibilité et l'évolutivité sont cruciales.

Exemples de bases de données NoSQL :

- **MongoDB** : Une base de données orientée documents.
- **Cassandra** : Conçue pour traiter des volumes élevés de données réparties sur plusieurs serveurs.

Avantages et défis du Big Data

Avantages du Big Data :

- **Amélioration des prises de décision** : Les entreprises peuvent analyser des ensembles de données massifs pour obtenir des informations précieuses et prendre des décisions plus éclairées [15].
- **Personnalisation des services** : En utilisant des algorithmes basés sur le Big Data, les entreprises peuvent personnaliser leurs offres pour chaque utilisateur.
- **Innovation et découverte** : Le Big Data ouvre de nouvelles perspectives en permettant de découvrir des modèles cachés dans des données massives [17].

Défis du Big Data :

- **Stockage et traitement** : Gérer d'énormes volumes de données nécessite des infrastructures coûteuses et complexes.
- **Véracité et qualité des données** : Toutes les données collectées ne sont pas toujours fiables ou pertinentes, ce qui complique l'analyse.

- **Protection de la vie privée** : Le traitement de grandes quantités de données sensibles soulève des préoccupations importantes en matière de sécurité et de confidentialité.

2.1.4 Reconnaissance Faciale

La reconnaissance faciale est un sous-domaine interdisciplinaire de la vision par ordinateur qui identifie le visage d'une personne en l'analysant à partir d'une image ou d'une vidéo à l'aide d'une caractéristique faciale particulière. Cette technologie fonctionne principalement avec des algorithmes d'apprentissage automatique et utilise actuellement des modèles d'apprentissage profond, comme les CNN. En outre, la reconnaissance faciale peut être utilisée à de nombreuses fins dans les applications de sécurité, de surveillance, d'authentification et de personnalisation des services [18].

Définition de la reconnaissance faciale

La reconnaissance faciale suit généralement trois étapes principales : la détection, la normalisation et la reconnaissance.

1. **Détection faciale** : L'algorithme localise un visage dans une image ou une vidéo.
2. **Normalisation et alignement** : Les visages détectés sont alignés et normalisés pour garantir que toutes les images sont comparées dans des conditions similaires.
3. **Reconnaissance faciale** : L'algorithme compare le visage avec une base de données de visages pour identifier ou vérifier l'identité de la personne.

Les méthodes modernes de reconnaissance faciale utilisent des modèles de Deep Learning, tels que les réseaux convolutifs (CNN), pour extraire des caractéristiques discriminantes et effectuer la classification des visages.

Types de reconnaissance faciale

Il existe plusieurs approches pour effectuer la reconnaissance faciale, en fonction des algorithmes utilisés et du type de données disponibles :

1. Reconnaissance faciale basée sur des caractéristiques géométriques

Cette méthode se concentre sur l'extraction des **points clés** du visage, comme la distance entre les yeux, la largeur du nez, la forme de la mâchoire, etc. Ces mesures géométriques sont ensuite utilisées pour créer une représentation unique du visage.

Caractéristiques :

- Méthode relativement simple et rapide.
- Moins efficace lorsque l'expression faciale ou l'éclairage change.

Exemples d'utilisation :

- Systèmes de surveillance de base.
- Accès sécurisé basé sur la reconnaissance géométrique simple.

Formule mathématique simple : La distance entre deux points A et B sur un visage peut être calculée à l'aide de la distance euclidienne :

$$d(A, B) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Cette formule est utilisée pour calculer des distances entre les yeux, la bouche, et d'autres points clés du visage.

2. Reconnaissance faciale basée sur des textures

Cette méthode repose sur l'analyse de la texture de la peau et des motifs du visage. Des algorithmes tels que les **histogrammes de gradients orientés** (HOG) ou les **transformées de Fourier** sont utilisés pour capturer les textures du visage, comme les rides, les pores, et autres détails fins qui peuvent différencier les visages.

Caractéristiques :

- Sensible aux variations d'éclairage et à l'angle de vue.
- Peut-être combinée avec d'autres méthodes pour améliorer la précision.

Exemple d'utilisation :

- Analyse détaillée dans les systèmes de sécurité de haut niveau.

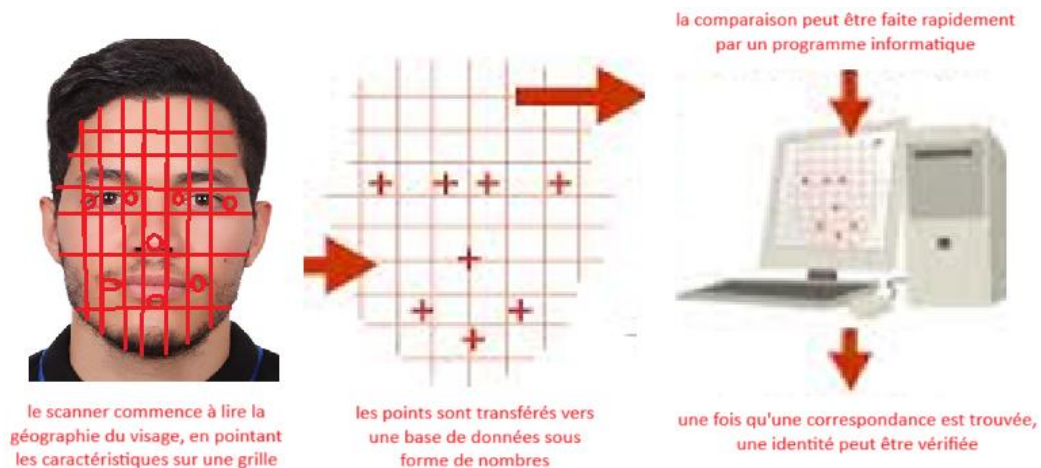


Figure 5 : Point clés d'un visage en reconnaissance faciale

3. Reconnaissance faciale avec Deep Learning (CNN)

La **reconnaissance faciale** moderne, lorsqu'elle devient la plus complexe et la plus précise, s'effectue grâce aux CNN. Ceux-ci permettent l'extraction automatique de caractéristiques hiérarchiques à partir d'images faciales sans ingénierie manuelle des caractéristiques. Les CNN analysent les images en plusieurs couches, chacune capturant des caractéristiques plus abstraites que la précédente.

Caractéristiques :

- Très efficace, notamment avec de grands ensembles de données.
- Capable de s'adapter à des variations d'éclairage, d'expression, et d'angle de vue.
- Utilisé dans des applications commerciales et gouvernementales de grande envergure.

Exemples d'utilisation :

- Authentification faciale dans les smartphones (Face ID d'Apple).
- Systèmes de surveillance avancés.

Les réseaux convolutifs traitent les images en appliquant des filtres pour extraire des caractéristiques discriminantes. Mathématiquement, l'opération de **convolution** est définie par :

$$S(i, j) = (I * K)(i, j) = \sum_m \sum_n I(i + m, j + n)K(m, n)$$

Où :

- $I(i, j)$ représente l'intensité du pixel de l'image à la position (i, j) ,
- $K(m, n)$ représente le noyau de convolution à la position (m, n) ,
- $S(i, j)$ est la valeur obtenue après convolution.

4. Reconnaissance faciale 3D

La **reconnaissance faciale 3D** utilise des capteurs pour capturer une représentation tridimensionnelle du visage. Contrairement à la reconnaissance 2D, qui repose uniquement sur des images planes, la reconnaissance 3D utilise des informations sur la profondeur et les contours du visage, ce qui la rend plus robuste aux variations d'éclairage et aux changements d'angle.

Caractéristiques :

- Capable de gérer les variations d'éclairage et d'angle de vue.
- Plus coûteux en termes de matériel (capteurs 3D).

Exemple d'utilisation :

- Systèmes de sécurité haut de gamme (scanners 3D dans les aéroports). (voir tableau 4)

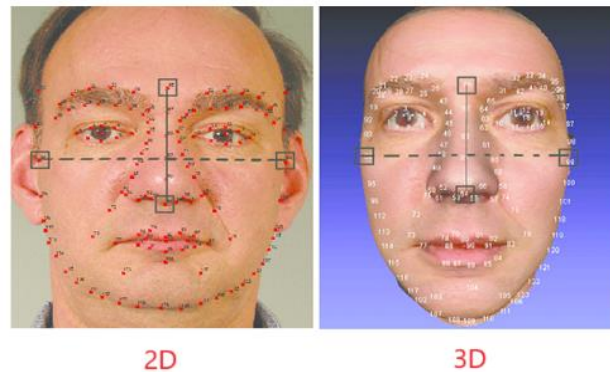


Figure 6 : Comparaison 2D vs 3D en reconnaissance faciale

Méthode	Caractéristiques principales	Avantages	Limites
Basée sur les caractéristiques géométriques	Utilise des mesures géométriques du visage	Rapide et simple	Sensible aux changements d'expression et d'éclairage
Basée sur les textures	Analyse les motifs de texture de la peau	Précise pour des visages sous conditions stables	Sensible aux variations d'éclairage et aux angles
Deep Learning (CNN)	Utilise des réseaux de neurones convolutifs pour extraire des caractéristiques	Très performant, s'adapte aux variations de visage	Nécessite un grand volume de données et des ressources de calcul

Reconnaissance faciale 3D	Utilise une représentation tridimensionnelle du visage	Robuste aux variations d'angle et d'éclairage	Coûts élevés, capteurs 3D nécessaires
----------------------------------	--	---	---------------------------------------

Tableau 4 : Comparaison des méthodes de reconnaissance faciale

Avantages et défis de la reconnaissance faciale

Avantages :

- **Précision** : Avec les technologies modernes de Deep Learning, la reconnaissance faciale atteint des niveaux de précision très élevés, même dans des environnements complexes [35].
- **Vitesse** : Les systèmes de reconnaissance faciale peuvent identifier ou vérifier une personne en temps réel.
- **Sans contact** : Contrairement à d'autres formes de biométrie, la reconnaissance faciale ne nécessite aucun contact physique.

Défis :

- **Protection de la vie privée** : L'utilisation croissante de la reconnaissance faciale, notamment dans les lieux publics, suscite des inquiétudes en matière de surveillance et de protection des données personnelles [18].
- **Biais algorithmiques** : Les systèmes de reconnaissance faciale peuvent montrer des biais raciaux ou de genre en raison de la composition des ensembles de données d'entraînement [19].
- **Fiabilité dans des environnements complexes** : La reconnaissance faciale peut être perturbée par des changements dans l'éclairage, des angles de vue extrêmes, ou l'utilisation de masques faciaux.

2.1.5 Impact du port de masques sur les systèmes de reconnaissance faciale

La pandémie de COVID-19 a complètement changé le contexte des systèmes de reconnaissance faciale. Le port généralisé de masques faciaux dans de nombreux pays présente de nouveaux défis pour le fonctionnement des systèmes de reconnaissance faciale, car ces masques couvrent la plupart des caractéristiques faciales clés, y compris la bouche et le nez.

Avant la pandémie, tous les systèmes de reconnaissance faciale étaient réglés pour identifier un visage complet. Les algorithmes ont été entraînés avec des images de visages où toutes les caractéristiques étaient visibles, ce qui permet d'obtenir des résultats précis. Cependant, les masques faciaux brisent cet équilibre en masquant jusqu'à 70 % des caractéristiques faciales.

Cela a entraîné une augmentation considérable du taux d'erreur des systèmes d'identification faciale. Des études ont montré que le port du masque peut réduire la précision de la reconnaissance faciale de plus de 50 %. Dans une étude, l'Institut national des normes et de la technologie des États-Unis a signalé que les masques affectaient les performances de la plupart des algorithmes commerciaux de reconnaissance faciale qui ont été testés, multipliant par cinq les taux d'erreur dans certains cas.

L'étude propose plusieurs solutions pour surmonter ce défi, comme par exemple réentraîner les algorithmes en incluant des images de visages masqués dans leurs jeux de données d'entraînement. D'autres chercheurs se concentrent sur la partie non masquée du visage, par exemple les yeux et le front, pour effectuer des identifications. Des approches basées sur l'apprentissage multitâche ont ainsi été développées, permettant de se concentrer sur les régions non masquées, en ignorant les parties masquées des visages.

Entre-temps, des systèmes sont adaptés pour combiner la reconnaissance faciale avec des technologies biométriques, telles que la reconnaissance de l'iris ou de la parole, afin de compenser les informations visuelles perdues avec les masques.

Toutefois, aucune de ces solutions n'est tout à fait sans problème. D'importants ajustements techniques sont nécessaires et leurs modèles devront être réentraînés à l'aide de nouvelles données. Des questions éthiques se posent également concernant la surveillance de masse et la manière dont elle pourrait s'améliorer grâce à l'acquisition de nouvelles données de visages

masqués, ce qui renforce les inquiétudes en matière de confidentialité et de sécurité des données.

2.2 PROBLÉMATIQUE ET JUSTIFICATION DE LA RECHERCHE

La reconnaissance faciale est une technologie qui a récemment fait l'objet d'une attention accrue, compte tenu d'une grande variété d'applications allant de la sécurité publique à la personnalisation des services. Cependant, la pandémie de COVID-19 a révélé des limites remarquables des systèmes de reconnaissance faciale existants, principalement en ce qui concerne l'impact des masques faciaux portés par une grande partie de la population mondiale. Bien qu'un masque soit nécessaire pour protéger la santé des personnes, il cache une grande partie du visage et rend les systèmes de reconnaissance faciale beaucoup moins efficaces, voire inopérants dans certaines situations. L'enjeu de ce défi concerne le fonctionnement du système de reconnaissance faciale, s'il était adapté à un visage partiellement masqué. Cela nécessite l'amélioration des modèles d'apprentissage automatique qui ont intégré des techniques pour gérer cette situation inhabituelle, car elle montre la dégradation des performances des différents algorithmes existants.

Cette étude s'appuie sur l'importance croissante des technologies de reconnaissance faciale, dans des domaines de plus en plus variés comme la sécurité, l'accès aux bâtiments, le contrôle biométrique et la nécessité de faire face aux nouvelles conditions imposées par la pandémie. Après une analyse approfondie et détaillée, l'élaboration de méthodes adaptées aux visages masqués améliorera non seulement les performances mais ouvrira également une nouvelle perspective dans la recherche sur l'apprentissage profond et le traitement d'images dans un environnement contraint.

2.3 OBJECTIFS ET DÉFIS DE RECHERCHE

L'objectif principal de cette recherche est de proposer une méthode efficace pour la reconnaissance faciale en présence de masques couvrant partiellement le visage. Pour atteindre cet objectif général, plusieurs sous-objectifs ont été identifiés :

1. **Analyser les limites des systèmes de reconnaissance faciale existants** face aux visages masqués, afin d'identifier les faiblesses spécifiques des algorithmes utilisés.

2. **Développer un modèle d'apprentissage profond** capable d'améliorer la précision de la reconnaissance faciale dans des environnements où le port du masque est requis.
3. **Évaluer l'impact des différentes caractéristiques faciales visibles** (telles que les yeux, le front, et les parties non masquées) pour déterminer les zones les plus utiles à la reconnaissance.
4. **Comparer l'efficacité des différentes bases de données** pour la reconnaissance de visages masqués et non masqués, afin de mieux comprendre les avantages et les limites des jeux de données existants.

Tout d'abord, ces objectifs sont associés à plusieurs défis techniques et méthodologiques. La construction ou l'adaptation de bases de données riches avec suffisamment d'exemples de visages masqués est l'un des défis majeurs. Par ailleurs, il sera nécessaire de concevoir des modèles capables de fonctionner efficacement avec des informations partielles tout en minimisant les erreurs de reconnaissance ou d'identification. Enfin, un défi très pertinent est l'évaluation des performances des modèles en conditions réelles.

2.4 APERÇU DE LA MÉTHODOLOGIE

La méthodologie adoptée dans cette recherche repose principalement sur l'utilisation d'algorithmes de deep learning, en particulier des réseaux de neurones convolutifs (CNN), pour traiter le problème de la reconnaissance faciale avec des visages masqués. Le processus méthodologique se décompose en plusieurs étapes clés :

1. **Collecte de données** : Utilisation de bases de données publiques et privées de visages masqués et non masqués, ainsi que création d'un ensemble de données spécifique pour la recherche.
2. **Prétraitement des données** : Nettoyage et préparation des images pour l'entraînement des modèles, incluant des techniques de normalisation et d'augmentation de données pour améliorer la robustesse du modèle.

3. **Modélisation** : Développement de réseaux de neurones convolutifs adaptés à la reconnaissance faciale partielle, en se concentrant sur l'extraction des caractéristiques des parties non masquées du visage.
4. **Entraînement et validation des modèles** : Utilisation de diverses méthodes de validation croisée pour évaluer la performance des modèles en termes de précision et de taux de fausses acceptations/fausses rejets.
5. **Comparaison de modèles** : Analyse comparative des performances entre différents modèles et algorithmes afin de déterminer la solution la plus adaptée.

2.5 CONCLUSION

Ces technologies vont du Deep Learning au Big Data en passant par la reconnaissance faciale et prouvent que la reconnaissance faciale masquée pose de nouveaux défis qui ne peuvent être surmontés que par une combinaison de ces technologies. Grâce à de grands ensembles de données, à une architecture de réseau neuronal avancée et à une évolution rapide des algorithmes, on est en mesure de proposer des solutions augmentant la précision et la robustesse des systèmes actuels. Cependant, la plupart des questions ouvertes concernent la gestion des biais algorithmiques et la protection de la vie privée.

TROISIÈME CHAPITRE - ÉTAT DE L'ART

3.1 INTRODUCTION

La justification des études ci-dessus, qui se rapprochent des mises en œuvre des mécanismes de reconnaissance faciale dans la période post-pandémique, ne peut être remise en cause, notamment parce que la technologie utilisée, la méthodologie et la stratégie en place sont toutes des tournants. L'objectif de cette étape est d'examiner les progrès réalisés dans les réseaux neuronaux convolutionnels (CNN), qui sont aujourd'hui la technologie de base des reconnaissances faciales et d'autres techniques de détection de visages et de masques. La brève analyse des approches déjà existantes aidera à situer cette recherche particulière dans un tableau scientifique plus large et ainsi à évaluer la faisabilité de tels systèmes de reconnaissance de masques. Le chapitre est divisé en plusieurs sections, la première étant l'analyse approfondie des réseaux neuronaux convolutionnels, qui sont principalement utilisés pour l'analyse d'images. Ensuite, dans ce chapitre, nous expliquerons ces techniques de détection de visages et soulignerai la manière dont ces méthodes sont utilisées pour détecter les caractéristiques des visages dans divers contextes. À la fin de ce chapitre, nous examinerons les développements avancés dans la détection de masques, qui a gagné en popularité en raison de la COVID-19. Ensuite, nous fournirons les modèles de reconnaissance faciale ainsi que la cartographie logique des structures typiques les plus courantes de ces modèles.

Ce chapitre va déterminer les forces et les faiblesses des méthodes existantes afin de fournir une meilleure structure de travail pour de nouvelles améliorations et recherches sur notre projet.

3.2 RÉSEAU DE NEURONES CONVOLUTIFS

Les tâches de vision par ordinateur utilisent généralement différents types de réseaux neuronaux, les réseaux neuronaux convolutionnels (CNN) étant considérablement popularisés pour les tâches orientées image. Ces dernières années, ces méthodes ont été appliquées avec succès à des problèmes de reconnaissance, tels que la reconnaissance faciale, la reconnaissance d'objets et la catégorisation d'images. Les CNN, qui ont émergé en 1989 grâce aux travaux de LeCun [12], peuvent donc être considérés comme un tournant dans l'évolution des modèles informatiques dans lesquels les images sont fournies et les caractéristiques apprises au lieu d'être conçues. Au cours de ce chapitre, nous examinerons la structure fine des réseaux convolutionnels et les moyens de son application dans le système de détection et de reconnaissance faciale. Nous décrirons également diverses structures, notamment VGGNet, ResNet et Facenet, qui sont reconnues comme des appareils professionnels quasi absolus pour la reconnaissance faciale.

Architecture des CNN

Un CNN est composé de plusieurs couches successives, chacune ayant une fonction spécifique dans le traitement des données. Ces couches comprennent des **couches de convolution**, des **couches de regroupement (pooling)**, des **couches entièrement connectées**, et une **couche de sortie**.

1. **Couches de convolution** : Les couches de convolution appliquent des filtres à l'image d'entrée pour extraire des caractéristiques locales. Ces filtres, ou noyaux, permettent de détecter des motifs dans l'image tels que des bords, des textures et d'autres formes visuelles importantes. Plus un réseau a de couches de convolution, plus il est capable de capturer des détails complexes à partir de l'image [12].
2. **Couches de regroupement (pooling)** : Après la convolution, les couches de regroupement sont utilisées pour réduire la dimensionnalité des données tout en conservant les informations les plus importantes. Cette réduction aide à rendre les réseaux plus efficaces tout en limitant le surapprentissage. Le *max pooling* est la méthode la plus courante, où seul le maximum d'une région est retenu pour les étapes suivantes [20].

3. **Couches entièrement connectées** : À ce stade, les caractéristiques extraites par les couches de convolution sont aplaties et transmises aux couches entièrement connectées, où chaque neurone est connecté à tous les neurones de la couche précédente. Ces couches permettent au réseau de prendre des décisions de classification basées sur les caractéristiques apprises.
4. **Couche de sortie** : Enfin, la couche de sortie produit la prédiction finale. Dans une tâche de classification binaire telle que la reconnaissance de visage masqué/non masqué, la sortie serait une probabilité indiquant si une image appartient à une certaine classe [21]. (voir tableau 5)

Fonctionnement des CNN

Le fonctionnement des CNN repose sur une série d'étapes d'apprentissage qui consistent à ajuster les poids des filtres à travers un processus d'optimisation. Le processus d'entraînement consiste en plusieurs phases :

1. **Convolution** : Les filtres convolutifs apprennent à détecter des motifs dans l'image d'entrée, comme les traits du visage, les contours, et les structures plus abstraites. Les CNN permettent d'extraire ces caractéristiques automatiquement sans avoir besoin d'une étape manuelle de sélection de caractéristiques [10].
2. **Activation** : La fonction d'activation *ReLU* (Rectified Linear Unit) est couramment utilisée dans les CNN pour introduire de la non-linéarité dans le modèle, ce qui permet au réseau de modéliser des relations complexes entre les pixels d'entrée.
3. **Pooling** : Le regroupement (ou *pooling*) permet de réduire la taille des cartes de caractéristiques tout en conservant les informations essentielles, rendant ainsi le modèle plus robuste aux petites variations dans l'image [20].
4. **Rétropropagation** : Les CNN sont entraînés à l'aide de l'algorithme de rétropropagation, qui ajuste les poids des filtres en fonction de l'erreur entre la prédiction et la vérité terrain. Les gradients sont calculés à chaque étape, et les poids sont mis à jour via des méthodes de descente de gradient telles que l'optimiseur Adam [22].

Applications des CNN dans la reconnaissance faciale

Les CNN sont largement utilisés pour la reconnaissance faciale, car ils peuvent extraire automatiquement les caractéristiques faciales les plus importantes à partir d'une image. Quelques applications spécifiques sont :

1. **Reconnaissance faciale dans les systèmes de sécurité** : Les CNN sont utilisés pour identifier des visages dans des flux vidéo en temps réel. Les réseaux formés sur de grandes bases de données de visages peuvent identifier des individus spécifiques, même dans des conditions variées, telles que des angles de vue différents ou des éclairages changeants [23].
2. **Déverrouillage de dispositifs personnels** : Les smartphones utilisent des CNN pour la reconnaissance faciale afin de déverrouiller les appareils de manière sécurisée. Des technologies comme Face ID d'Apple utilisent des modèles CNN pour détecter et reconnaître les visages dans des conditions d'utilisation quotidiennes [24].
3. **Vérification d'identité en ligne** : Les systèmes bancaires et de vérification d'identité en ligne intègrent des CNN pour s'assurer que les utilisateurs sont bien ceux qu'ils prétendent être, en comparant les images en temps réel avec des images de référence [23].
4. **Détection de masques faciaux** : Les CNN ont été utilisés pour la détection des masques faciaux, une application devenue cruciale pendant la pandémie de COVID-19. Des systèmes peuvent désormais détecter automatiquement si une personne porte un masque et si le masque couvre correctement le nez et la bouche [25].

Technologie	Année d'introduction	Caractéristiques clés	Applications	Avantages	Inconvénients
CNN	1998	Couches convolutives simples avec pooling, fully connected layers.	Classification d'images, détection d'objets, reconnaissance faciale.	Facile à mettre en œuvre, efficace pour les petites tâches d'image.	Moins performant pour les réseaux très profonds, surapprentissage possible.
VGGNet	2014	Couches convolutives plus profondes avec des filtres 3x3, plus coûteux en calcul.	Classification d'images à haute performance.	Excellente extraction des caractéristiques profondes.	Prend beaucoup de mémoire et de puissance de calcul.
ResNet	2016	Utilisation de connexions résiduelles pour permettre des réseaux très profonds.	Reconnaissance d'images à grande échelle.	Permet des réseaux très profonds sans dégradation de performance.	Complexité accrue, exige plus de ressources de calcul.
FaceNet	2015	Utilise <i>triplet loss</i> pour apprendre des représentations d'images de visages.	Reconnaissance et clustering des visages.	Très précis dans les tâches de reconnaissance faciale.	Nécessite de grandes bases de données et une forte puissance de calcul.

Tableau 5 : Comparaison des architectures CNN

Défis et Limites des CNN

Bien que les CNN aient considérablement amélioré la reconnaissance faciale, plusieurs défis subsistent :

1. **Dépendance aux données** : Les CNN nécessitent des ensembles de données massifs pour bien généraliser. Si les données d'entraînement sont biaisées, les résultats du modèle le seront également, comme l'ont montré des études sur les biais raciaux et de genre dans les systèmes de reconnaissance faciale [19].
2. **Coût en calcul** : Les réseaux convolutifs, en particulier les versions plus profondes comme ResNet et VGGNet, exigent des ressources de calcul importantes, notamment pour l'entraînement sur de grandes images ou vidéos en haute résolution.
3. **Problèmes de sécurité et de confidentialité** : L'utilisation des CNN pour la reconnaissance faciale soulève des préoccupations majeures en matière de confidentialité et de sécurité. Des attaques adversariales peuvent tromper les systèmes de reconnaissance faciale en introduisant de petites perturbations dans l'image [6].

3.3 TECHNIQUES DE DÉTECTION DE VISAGES

Le processus de reconnaissance faciale comporte de nombreuses étapes, dont la détection faciale. En plus de permettre au système d'identifier le visage dans une image ou une vidéo, il comprend également la détermination des positions des groupes de visages et la séparation des visages afin que la reconnaissance puisse avoir lieu ultérieurement. Pour cette raison, cette section présente trois des approches qui ont été développées au fil des ans pour améliorer l'efficacité de la détection en fonction de l'évolution de l'éclairage, de l'angle et de la qualité de la photographie ou de la vidéo. Au cours de cette section, quatre méthodes importantes qui ont pu améliorer considérablement le secteur seront examinées : **l'algorithme de Viola et Jones, les modèles binaires locaux (LBP), la modélisation par mélange gaussien (GMM) et les stratégies basées sur l'apprentissage profond (CNN)**. Les détails de chacune des techniques seront fournis à tour de rôle et leur interrelation en termes d'avantages, d'inconvénients et de situations d'utilisation possibles seront analysés.

Algorithme de Viola-Jones (2001)

L'algorithme de Viola-Jones est l'une des méthodes pionnières de détection de visage qui a gagné en popularité et a trouvé des applications dans les systèmes en temps réel. Cet algorithme utilise la détection de visage basée sur les caractéristiques de **Haar** ainsi que la technique **AdaBoost** pour sélectionner les classificateurs faibles. Il intègre également une structure en cascade qui améliore la vitesse en éliminant la plupart des zones non pertinentes de l'image à chaque étape.

Haar Cascade :

Haar Cascade est une technique de détection d'objets développée par **Paul Viola** et **Michael Jones** en 2001 [26], marquant une avancée significative dans la vision par ordinateur, en particulier dans la reconnaissance faciale. Elle utilise une méthode qui exploite les caractéristiques de Haar dans une architecture en cascade pour améliorer la vitesse de traitement des images. Le terme « **Haar** » est dérivé des ondes de **Haar**, qui sont des constructions mathématiques qui analysent les variations d'intensité des pixels dans une image. Cette approche a marqué un tournant dans la détection en temps réel, car elle permettait de détecter des objets, en particulier des visages, avec une grande rapidité et précision, tout en étant suffisamment légère pour fonctionner sur des systèmes à faible puissance de calcul.

1. Caractéristiques de Haar

Les caractéristiques Haar sont essentiellement un type de filtre utilisé dans le traitement d'images qui met en évidence les variations d'intensité lumineuse entre des zones carrées ou rectangulaires voisines, telles que les bords, les contours et la texture. Chaque filtre se compose d'un motif simple et discret qui évalue la différence d'intensité blanche ou noire de ces zones rectangulaires. Les filtres Haar peuvent fournir efficacement des informations structurelles dans les images faciales, notamment :

- Les **contours des yeux** (détection d'une différence d'intensité entre les zones sombres des yeux et le front),
- Les **bords du nez** (avec des variations d'intensité le long des arêtes),

- **La zone de la bouche.**

Chaque caractéristique de Haar est appliquée sur des sous-régions d'une image, et un score est calculé pour chaque région en fonction de la différence d'intensité entre ces sous-régions.

Exemple de caractéristiques de Haar :

Les caractéristiques de Haar peuvent prendre différentes formes, comme illustré ci-dessous :

- **Deux rectangles adjacents** : Un rectangle clair et un rectangle sombre. Si le rectangle clair a une intensité plus élevée que le sombre, une forte réponse est générée.
- **Trois rectangles** : Utilisé pour capturer des changements brusques d'intensité, comme au niveau du nez.
- **Quatre rectangles** : Utilisé pour les régions complexes avec des variations d'intensité, telles que la bouche ou les yeux.

Illustration suggérée : Un exemple de différents types de filtres de Haar appliqués à une image faciale, avec des zones claires et sombres pour détecter les variations.

2. Cascade de classificateurs

L'innovation majeure de la méthode Haar Cascade réside dans son approche en **cascade de classificateurs**, qui permet d'accélérer considérablement la détection. Plutôt que d'analyser chaque pixel d'une image avec des filtres complexes, Viola et Jones ont proposé une cascade de décisions progressives, où chaque étape rejette rapidement les zones qui ne contiennent probablement pas de visages. (voir figure 7)

Cette cascade fonctionne de la manière suivante :

- **Étape 1** : Un classificateur simple applique un filtre de Haar sur une sous-région de l'image. Si la région échoue à ce test, elle est rejetée immédiatement.

- **Étape 2** : Si la région réussit le test, un classificateur plus complexe analyse la région avec un ensemble plus sophistiqué de filtres de Haar.
- **Étapes successives** : Chaque étape applique des tests de plus en plus complexes, ce qui réduit progressivement le nombre de régions candidates tout en augmentant la précision.

L'idée clé est que les étapes initiales du classificateur sont très rapides et rejettent la majorité des régions non pertinentes. Seules les régions qui ont passé tous les tests de la cascade sont identifiées comme contenant des visages.

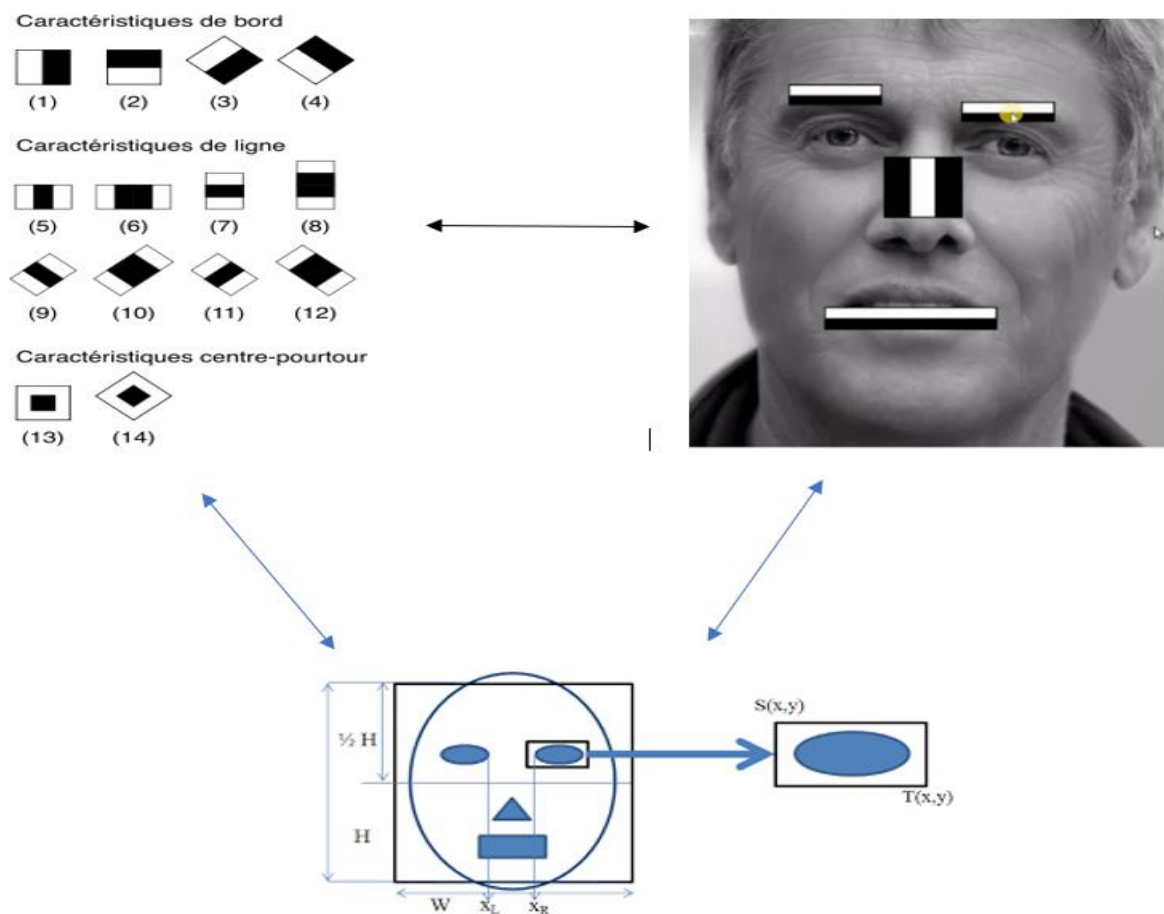


Figure 7 : Représentation de filtrage successif de la méthode Haar cascade

3. Avantages et limites du Haar Cascade

Avantages :

1. **Vitesse** : Le principal avantage de l'approche Haar Cascade est sa capacité à détecter rapidement des visages dans des images, même sur des systèmes avec des ressources limitées. Cela en fait une solution idéale pour les applications en temps réel, telles que la vidéosurveillance.
2. **Simplicité** : Les caractéristiques de Haar sont relativement simples à calculer, ce qui rend cette méthode efficace sur des images de différentes résolutions.
3. **Précision raisonnable** : Malgré son âge, l'algorithme de Viola-Jones [26] reste relativement précis dans des conditions de détection simple (comme des visages bien éclairés, de face).

Limites :

1. **Sensible aux variations** : L'un des principaux inconvénients du Haar Cascade est sa sensibilité aux variations d'éclairage, d'angle de vue, et aux expressions faciales. Si un visage n'est pas parfaitement orienté vers la caméra, la précision de la détection diminue.
2. **Précision limitée dans les environnements complexes** : Dans des environnements complexes ou avec des visages partiellement masqués, les performances chutent considérablement.
3. **Dépendance aux images frontales** : Les performances de détection se dégradent fortement pour les visages non frontaux ou avec des occlusions importantes.

Caractéristiques locales binaires (LBP)

Les **caractéristiques locales binaires (Local Binary Patterns, LBP)** sont une approche basée sur l'analyse des textures locales d'une image [27]. L'idée est de diviser une image en petits blocs et d'analyser les textures de chaque bloc à l'aide d'un opérateur LBP. Cette technique

encode la relation entre un pixel central et ses voisins, ce qui produit un motif binaire représentatif de la texture locale.

Caractéristiques principales :

- Robuste aux variations d'éclairage grâce à sa représentation basée sur les textures.
- Simplicité d'implémentation et rapidité de calcul.
- Utilisation dans la détection de visages, en particulier dans des environnements à faible contraste [28].

Limites :

- Moins performant pour des visages sous des angles non frontaux.
- Peut perdre en précision dans des environnements complexes ou avec des occlusions [18].

Modèles de mélange gaussien (GMM)

Les modèles de mélange gaussien (GMM) sont utilisés pour modéliser les distributions probabilistes de l'apparence du visage dans une image [18]. Chaque visage est considéré comme un mélange de composants de plusieurs distributions gaussiennes, qui capturent la structure de l'apparence de différentes régions d'un visage (par exemple, les yeux, le nez et la bouche). En plaçant des hypothèses dans le modèle gaussien, il peut devenir possible de détecter le même visage sous différents éclairages et angles.

Caractéristiques principales :

- Approche probabiliste qui permet de modéliser les variations de forme et de texture des visages.
- Robustesse face aux changements de lumière et d'angle.
- Convient bien aux environnements contrôlés (comme des systèmes de surveillance) [18].

Limites :

- Sensible aux occlusions partielles.
- Le calcul des paramètres peut être coûteux en temps et en ressources.

Approches basées sur le Deep Learning (2015 - présent)

Les approches modernes de détection de visages reposent sur des réseaux de neurones convolutifs (CNN), qui surpassent largement les méthodes traditionnelles en termes de précision et de robustesse. Un des travaux pionniers dans ce domaine est celui de **Sun et al [30]. (2014)**, qui a montré que les CNN pouvaient être utilisés pour détecter des visages sous divers angles et conditions d'éclairage, avec une très grande précision.

1. **Faster R-CNN** : Cette méthode utilise un réseau de propositions régionales (Region Proposal Network, RPN) pour générer des régions d'intérêt où des visages sont susceptibles d'apparaître, avant de les classifier avec un CNN. Faster R-CNN est l'une des méthodes les plus précises, mais elle est plus lente que d'autres approches [29].
2. **YOLO (You Only Look Once)** : YOLO est une méthode de détection en temps réel qui divise l'image en une grille et prédit simultanément les boîtes englobantes et les probabilités pour chaque grille. Elle est extrêmement rapide et adaptée aux systèmes de surveillance en temps réel.

Caractéristiques principales :

- Très performant dans des conditions variées, même pour des visages masqués.
- Capable de gérer de grands ensembles de données et de détecter des visages en temps réel.
- Moins sensible aux changements d'angle et d'éclairage que les méthodes traditionnelles.

Limites :

- Exige une grande puissance de calcul, surtout pour l'entraînement.

- Peut-être plus coûteux à déployer dans des environnements où les ressources GPU (unité de traitement graphique) sont limitées. (voir tableau 6)

Méthode	Précision	Rapidité	Robustesse	Environnement optimal
Viola-Jones (2001)	Moyenne	Très rapide	Sensible aux variations	Systèmes de surveillance basique
LBP (2004)	Bonne	Rapide	Sensible aux angles	Environnements à faible contraste
Modèles GMM (2004)	Bonne	Modérée	Bonne, mais lente	Environnements contrôlés
CNN (2015 - présent)	Très élevée	Rapide (en temps réel)	Très robuste	Systèmes de surveillance avancés, vidéosurveillance en temps réel

Tableau 6 : Comparaison des techniques de détection de visages

Avantages et limites des méthodes modernes

Par rapport aux méthodes traditionnelles comme Viola-Jones [26] ou les caractéristiques binaires locales (LBP), les approches modernes, notamment celles comme les techniques d'apprentissage profond (par exemple, les CNN), présentent des performances nettement améliorées en termes de précision, de robustesse et d'adaptabilité aux situations difficiles. Les dernières méthodes de détection des visages utilisent des systèmes d'apprentissage profond pour s'adapter aux variations drastiques des angles d'inclinaison du visage, des conditions d'éclairage et même des expressions faciales. De telles capacités permettent souvent à ces

nouvelles procédures de surpasser les systèmes traditionnels de détection des visages dans les applications de vidéosurveillance, d'identification biométrique et de sécurité/sortie.

Avantages des méthodes modernes :

1. **Précision accrue** : Grâce à leur capacité à apprendre directement à partir des données, les modèles basés sur le Deep Learning atteignent des taux de précision largement supérieurs aux méthodes classiques. Ils sont capables de capturer des caractéristiques abstraites des visages, y compris des détails subtils qui seraient négligés par des méthodes basées sur des caractéristiques manuelles [30].
2. **Robustesse aux variations** : Contrairement aux méthodes classiques qui sont souvent sensibles aux changements d'éclairage ou d'angle de vue, les CNN sont capables de généraliser à des conditions variées grâce à leur capacité à extraire des représentations hiérarchiques. Ils fonctionnent bien même avec des visages partiellement obstrués ou sous différents angles.
3. **Adaptation en temps réel** : Des modèles comme **YOLO** et **Faster R-CNN** permettent une détection en temps réel, ce qui est crucial pour des applications de sécurité et de surveillance où des décisions rapides doivent être prises.
4. **Capacité à évoluer** : Les systèmes basés sur le Deep Learning peuvent être continuellement améliorés et réentraînés sur de nouvelles données, ce qui permet de suivre l'évolution des besoins et des conditions réelles [31].

Limites des méthodes modernes :

1. **Ressources nécessaires** : Les modèles de Deep Learning, en particulier ceux utilisés pour la détection de visages, nécessitent une **grande puissance de calcul** pour être entraînés, en raison de la taille et de la complexité des réseaux neuronaux. Ils dépendent souvent de GPU coûteux et d'infrastructures massivement parallèles pour un entraînement efficace, ce qui peut poser un problème dans des environnements à ressources limitées [29].

2. **Exigences en matière de données** : Ces modèles nécessitent des ensembles de données massifs et diversifiés pour être bien entraînés. Un manque de données ou des données biaisées peuvent nuire à la capacité du modèle à généraliser correctement, notamment dans le cas de groupes ethniques sous-représentés ou de contextes spécifiques, comme le port de masques [19].
3. **Sensibilité aux masques faciaux** : Bien que les méthodes modernes aient démontré une grande robustesse, le port de **masques faciaux** a révélé des faiblesses dans les systèmes de détection actuels. Ces derniers peinent souvent à identifier des visages partiellement obstrués, car une grande partie des caractéristiques distinctives du visage (la bouche, la mâchoire, le nez) est masquée. L'amélioration de la précision dans ces cas reste un défi actif dans le domaine [32].
4. **Coût de déploiement** : En plus du coût d'entraînement, le déploiement de ces systèmes dans des environnements réels, tels que les aéroports ou les installations de surveillance à grande échelle, peut être coûteux en termes de maintenance et de mise à jour des infrastructures matérielles.

3.4 MÉTHODES DE DÉTECTION DE MASQUES

L'apparition de la pandémie de COVID-19 a entraîné l'utilisation généralisée de masques faciaux dans de nombreux lieux publics, posant un nouveau problème aux systèmes de reconnaissance faciale. Les masques cachent une partie importante des traits du visage observables, dégradant les performances des différents systèmes existants. Afin de résoudre les problèmes causés par la pandémie, **la détection de masques faciaux** est devenue un nouveau domaine d'étude en vision par ordinateur, permettant à un système de déterminer si une personne porte un masque afin d'adapter la reconnaissance faciale au nouvel environnement de port du masque facial.

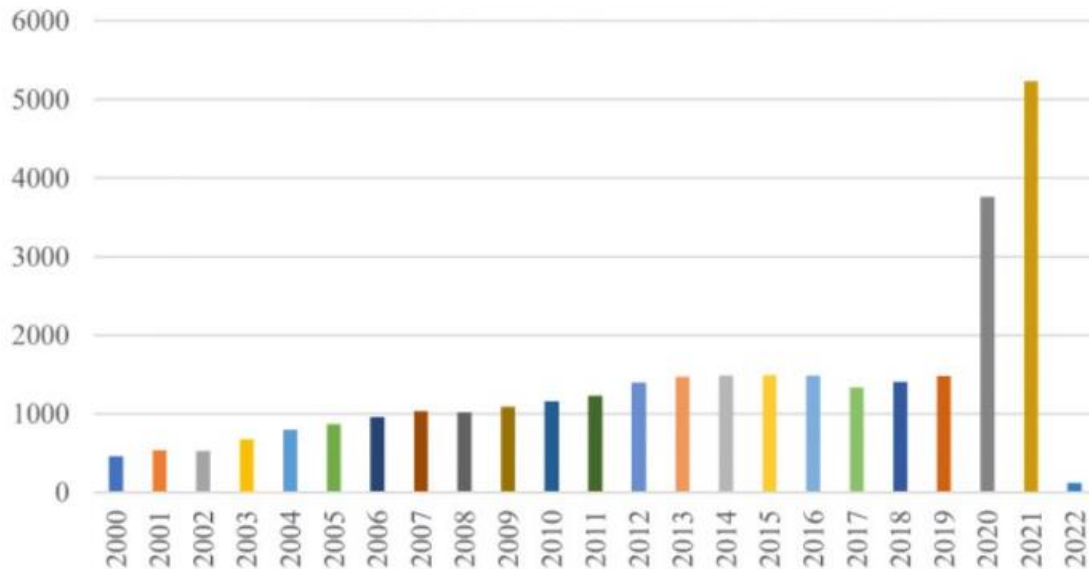


Figure 8 : évolution des efforts de recherche en détection de masques [32]

Importance de la détection de masques

La détection de masques joue un rôle crucial dans de nombreuses applications, notamment :

1. **Surveillance et sécurité publique** : Permet de surveiller efficacement des foules pour s'assurer que les directives de santé publique sont respectées.
2. **Systèmes de contrôle d'accès** : Ces systèmes peuvent ajuster leurs algorithmes de reconnaissance faciale en fonction de la présence ou non d'un masque pour garantir l'identification correcte des individus.
3. **Aide à la reconnaissance faciale** : En détectant d'abord si un masque est porté, les systèmes peuvent orienter leur attention vers des parties spécifiques du visage qui ne sont pas obstruées, comme les yeux ou le front.

Approches classiques de la détection de masques

Les premiers efforts visant à identifier l'utilisation de masques faciaux ont fait appel à des approches dérivées de descripteurs définis à la main, tels que les histogrammes de gradients orientés (HOG) et les modèles binaires locaux (LBP), couplés à des classificateurs tels que les machines à vecteurs de support (SVM). Ces approches ont obtenu un certain succès dans la

détection d'objets simples, mais ont jusqu'à présent été limitées en ce qui concerne la complexité et la variabilité associées aux masques faciaux (comme la couleur, la forme et la texture).

Méthode des histogrammes de gradients orientés (HOG)

Les informations sur les contours et les gradients d'une image peuvent être capturées grâce à des **histogrammes de gradients orientés, ou HOG**. La logique derrière cette approche est que les orientations locales des gradients peuvent capturer la forme et les contours des objets, comme les masques faciaux. Tout d'abord, nous traçons les gradients sur les images au sein des cellules, les descripteurs HOG pour chaque cellule sont ensuite calculés et utilisés pour classer l'image comme ayant un masque facial ou non.

Faiblesses : Bien qu'il induise une robustesse dans certaines situations, HOG est limité à une gamme étroite de variations de masques en taille, couleur et texture et il est sensible aux changements d'éclairage [33].

Approches modernes basées sur le Deep Learning

Cependant, la technique contemporaine utilisant le Deep Learning a été beaucoup plus efficace dans la détection de masques en raison de son apprentissage automatique des caractéristiques des données. **Les réseaux de neurones convolutionnels (CNN)**, ainsi que certaines architectures avancées, telles que YOLO (You Only Look Once) et Faster R-CNN, nous permettent de nous fier largement à la reconnaissance d'images sur la détection de masques.

1. Réseaux de neurones convolutionnels (CNN)

Les réseaux de neurones convolutionnels (**CNN**) se sont avérés très efficaces dans la détection de masques faciaux en raison de leur représentation hiérarchique des images. Un modèle pré-entraîné tel que ResNet ou MobileNetV2 peut être utilisé pour réaliser la détection de masques

sur un visage, malgré des conditions défavorables telles que des angles de vue compliqués ou des masques de formes différentes.

Processus de détection : Un modèle CNN a été préalablement formé sur un ensemble de données d'images avec des images faciales sans aucun masque. Au cours du processus d'inférence, le modèle peut parcourir une image faciale, qui encode ensuite l'image et passe par un certain nombre de convolutions qui produisent les caractéristiques encodées que nous voulons extraire.

Exemples d'utilisation :

- **FastMaskNet** : Un modèle basé sur des CNN conçu pour détecter les masques dans des environnements de surveillance en temps réel [34].

2. YOLO (You Only Look Once)

YOLO est un algorithme de détection d'objets en temps réel qui divise l'image en une grille et applique des prédictions de boîtes englobantes et de probabilités sur chaque région de la grille. Il est bien adapté pour la détection de masques faciaux, car il est capable de traiter des images en une seule passe avec une grande vitesse. YOLO est souvent utilisé dans des applications de vidéosurveillance où une détection rapide des masques est nécessaire. **(Voir figure 9)**

Avantages :

- **Vitesse** : YOLO est extrêmement rapide et capable de détecter des masques en temps réel, ce qui le rend idéal pour des environnements à haute performance comme les systèmes de contrôle d'accès.
- **Précision** : Avec des versions améliorées comme **YOLOv3** et **YOLOv4**, les performances en termes de précision sont également élevées [31].

Limites :

- Moins performant pour détecter des masques faciaux partiels ou des objets très petits dans l'image.

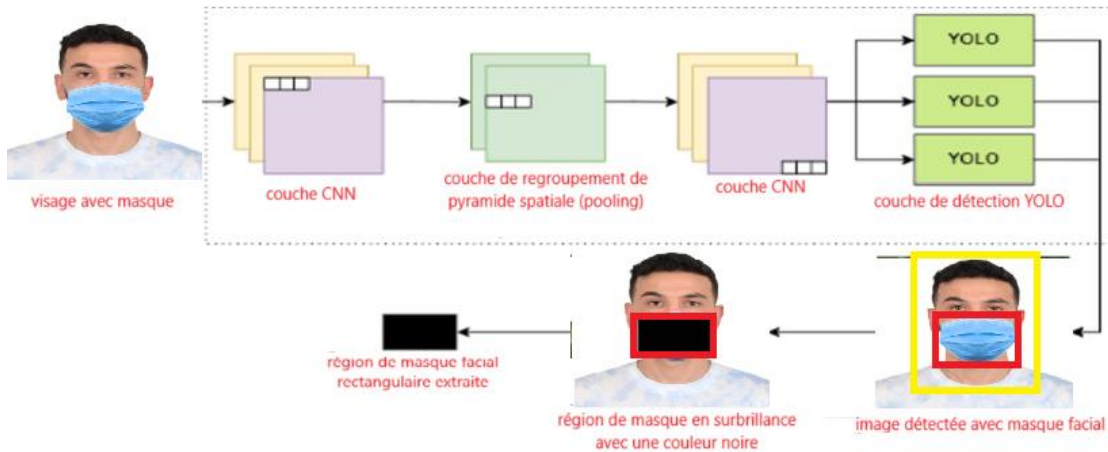


Figure 9 : Diagramme YOLO en détection d'image [31]

3. Faster R-CNN

Faster R-CNN est une autre méthode avancée qui utilise un **Réseau de Propositions Régionales (RPN)** pour générer des régions d'intérêt dans une image avant de les passer à travers un CNN pour la classification. Cette méthode offre une précision élevée pour la détection de masques, même dans des conditions de variation d'angle ou d'occlusion. (Voir figure10)

Avantages :

- Très précis pour la détection d'objets comme les masques, surtout dans des environnements complexes.
- Capable de traiter des images contenant plusieurs objets et masques.

Limites :

- Moins rapide que YOLO pour des applications en temps réel, mais plus précis [29].

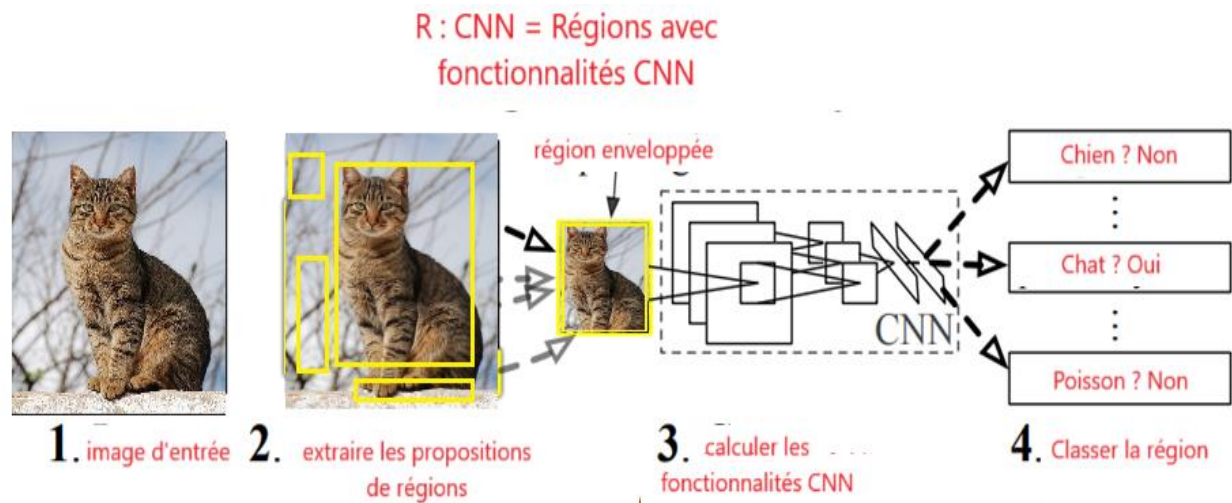


Figure 10 : Diagramme Faster R-CNN en détection d'objet [29]

Méthode	Précision	Rapidité	Robustesse	Environnement optimal
HOG + SVM	Moyenne	Modérée	Faible	Conditions contrôlées, lumière constante
CNN (ResNet, MobileNetV2)	Très élevée	Rapide	Très robuste	Surveillance, contrôle d'accès
YOLO	Bonne	Très rapide	Robuste	Surveillance en temps réel
Faster R-CNN	Très élevée	Modérée	Très robuste	Environnements complexes avec occlusions

Tableau 7 : Comparaison des méthodes de détection de masques

3.5 MODELES LES PLUS UTILISÉS EN RECONNAISSANCE FACIALE

La reconnaissance faciale est devenue un élément important dans divers domaines, notamment la sécurité, la surveillance et l'authentification biométrique. De nos jours, la plupart des systèmes de reconnaissance faciale sont basés sur des méthodes d'apprentissage profond et d'autres algorithmes sophistiqués capables d'extraire des caractéristiques uniques des visages humains et de les faire correspondre à de grands ensembles de données. Ainsi, dans cette section, nous passons brièvement en revue les modèles de reconnaissance faciale les plus courants, leurs configurations, leurs points forts et leurs points faibles.

1 DeepFace (2014)

DeepFace, développé par Facebook en 2014, est l'un des premiers modèles de reconnaissance faciale à démontrer une précision proche du niveau humain. Il repose sur un réseau de neurones convolutifs (CNN) profond, capable de capturer des informations tridimensionnelles des visages et de les utiliser pour la reconnaissance. **DeepFace** a atteint un taux de précision de 97,35 % sur la base de données **Labeled Faces in the Wild (LFW)**, un jalon important dans l'évolution de la reconnaissance faciale [35].

Caractéristiques :

- Utilisation d'un CNN profond pour l'extraction des caractéristiques.
- Conversion des images faciales en modèles 3D pour améliorer la robustesse aux variations d'angle et d'expression faciale.
- Utilisation de plus de 120 millions de paramètres pour capturer des informations fines du visage.

Avantages :

- Très performant pour les visages frontaux et sous différents angles.
- Robuste aux variations d'expression faciale et aux conditions d'éclairage.

Limites :

- Modèle volumineux nécessitant une grande puissance de calcul pour l'entraînement.

- Peut rencontrer des difficultés avec les visages masqués ou partiellement obstrués.

2 FaceNet (2015)

Créé par Google en 2015, **FaceNet** est l'un des modèles de reconnaissance faciale les plus connus. Au lieu d'utiliser une approche de classification des visages, **FaceNet** envisage une manière innovante d'apprendre un espace d'intégration dans lequel les visages similaires sont proches les uns des autres et les visages différents sont plus éloignés. L'algorithme est entraîné à l'aide d'une perte de triplet qui minimise la distance entre les images de la même personne et maximise la distance lorsque les visages sont différents [24].

Caractéristiques :

- Utilisation de la **triplet loss** pour apprendre un espace d'incorporation.
- Capable de générer un vecteur de 128 dimensions pour chaque visage, utilisé pour la comparaison de similarité.
- Très léger et efficace pour des systèmes de reconnaissance faciale en temps réel.

Avantages :

- Haute précision avec un taux de 99,63 % sur le dataset LFW.
- Modèle léger et adapté aux environnements nécessitant une faible latence, comme l'authentification biométrique sur smartphones.

Limites :

- La performance peut être affectée par des visages partiellement obstrués ou masqués.
- Nécessite de grands ensembles de données pour un entraînement efficace.

3 VGG-Face (2015)

VGG-Face, développé par l'Université d'Oxford, est un modèle de reconnaissance faciale basé sur un réseau de neurones convolutif profond similaire à l'architecture VGG utilisée pour la

classification d'images. **VGG-Face** se distingue par son architecture simple mais efficace, composée de 16 couches, et son utilisation d'images de haute résolution pour capturer des détails fins du visage [23].

Caractéristiques :

- 16 couches convolutives, avec un total de 138 millions de paramètres.
- Entraîné sur une base de données contenant plus de 2,6 millions de visages appartenant à plus de 2600 individus.
- Capable d'extraire des caractéristiques fines et détaillées des visages.

Avantages :

- Excellente précision sur les visages frontaux et bien alignés.
- Modèle bien adapté aux environnements où la résolution des images est élevée, comme les systèmes de vidéosurveillance.

Limites :

- Lourd en termes de paramètres, nécessitant des ressources matérielles considérables.
- Moins performant dans des conditions d'éclairage difficiles ou avec des visages non alignés.

4 ArcFace (2019)

ArcFace a été proposé par Deng et al [36]. En 2019 comme l'un des modèles les plus performants en matière de reconnaissance faciale à ce jour. Une perte angulaire, Arc Margin Loss, est proposée pour maximiser la séparation des différentes classes dans l'espace d'intégration pour la reconnaissance faciale, ce qui augmente considérablement la précision. **ArcFace** atteint une précision de plus de 99,8 % sur la base de données LFW [36].

Caractéristiques :

- Utilisation de la **Arc Margin Loss** pour améliorer la séparation des classes dans l'espace d'incorporation.

- Très robuste aux variations d'éclairage, d'angle et d'expression faciale.
- Architecture légère, bien adaptée aux systèmes de reconnaissance faciale en temps réel.

Avantages :

- Haute précision même dans des conditions difficiles, telles que des visages sous différents angles ou dans des environnements faiblement éclairés.
- Bien adapté pour des applications commerciales et gouvernementales nécessitant une grande précision.

Limites :

- Performances légèrement réduites lorsque les visages sont masqués ou partiellement obstrués.
- Nécessite de grands ensembles de données pour être entraîné efficacement.

5 Dlib (2015)

Dlib est l'une des bibliothèques open source les plus populaires liées à la reconnaissance faciale. De plus, Dlib propose un modèle de reconnaissance faciale basé sur un descripteur vectoriel à 128 dimensions. Bien que Dlib ne soit pas aussi puissant que certains autres modèles basés sur l'apprentissage profond, il reste actuellement dans le champ d'utilisation des environnements avec des limites de ressources matérielles, comme dans les applications mobiles et les systèmes embarqués.

Caractéristiques :

- Utilisation d'un descripteur de 128 dimensions pour chaque visage, similaire à **FaceNet**.
- Léger et facilement déployable sur des systèmes à faible puissance de calcul.
- Très populaire dans les projets open-source et les systèmes de reconnaissance faciale à petite échelle.

Avantages :

- Très léger et facile à déployer.
- Bonne performance pour des applications nécessitant une reconnaissance faciale basique ou des projets open-source.

Limites :

- Moins précis que les modèles modernes basés sur le Deep Learning.
- Sensible aux variations d'angle et d'éclairage.

6 OpenFace (2016)

OpenFace est une autre bibliothèque open-source qui s'inspire de **FaceNet** pour la reconnaissance faciale. Elle est conçue pour être légère, modulaire et facile à utiliser, ce qui en fait une option populaire pour des applications de reconnaissance faciale à faible coût.

OpenFace est utilisé dans des systèmes nécessitant une bonne précision sans les exigences en ressources matérielles des modèles plus complexes.

Caractéristiques :

- Basé sur un modèle d'apprentissage d'incorporation (embedding) similaire à **FaceNet**.
- Conçu pour être léger et facilement déployable dans des environnements à faible latence.

Avantages :

- Open-source et facilement personnalisable.
- Bonne performance dans des environnements où des ressources limitées sont disponibles.

Limites :

- Moins précis que des modèles comme **ArcFace** ou **DeepFace**.
- Moins robuste face aux défis tels que les visages masqués ou dans des environnements faiblement éclairés. (**Voir tableau 8**)

Modèle	Précision	Rapidité	Robustesse	Environnement optimal
DeepFace	Très élevée	Modérée	Haute	Réseaux sociaux, surveillance
FaceNet	Très élevée	Rapide	Haute	Authentification biométrique, mobile
VGG-Face	Très élevée	Modérée	Moyenne	Vidéosurveillance avec haute résolution
ArcFace	Très élevée	Rapide	Très haute	Applications commerciales, gouvernements
Dlib	Moyenne	Très rapide	Faible à modérée	Projets open-source, petits systèmes
OpenFace	Moyenne	Rapide	Moyenne	Systèmes mobiles, environnements à faible coût

Tableau 8 : Comparaison des modèles de reconnaissance faciale

Le modèle de reconnaissance faciale a connu un développement rapide au cours des dix dernières années, au cours desquelles certains modèles représentatifs tels que **DeepFace**, **FaceNet** et **ArcFace** ont atteint des performances de niveau humain dans un environnement complexe. Par rapport aux approches traditionnelles, les modèles basés sur l'apprentissage profond sont très en avance, mais chaque modèle présente des avantages spécifiques en fonction des besoins de vitesse, de précision et de robustesse.

3.6 CONCLUSION

Ce troisième chapitre a donné un aperçu de certaines techniques et modèles de pointe en matière de détection et de reconnaissance faciale, soulignant l'évolution des méthodes classiques vers des approches basées sur le **Deep Learning**. Ainsi, les réseaux de neurones convolutifs ont remplacé de loin les méthodes traditionnelles de détection ou de reconnaissance faciale, comme l'**algorithme de Viola-Jones** ou les fonctionnalités de **Haar**, et ont fourni plus de précision et une bien meilleure robustesse face aux changements d'angle, d'éclairage et d'expression.

Les méthodes de détection de masques sont également devenues très importantes dans le contexte de la pandémie mondiale, avec des algorithmes comme YOLO, Faster R-CNN ou des modèles CNN spécialisés qui ont montré leur efficacité pour répondre à ce nouveau défi. En parallèle, les modèles de reconnaissance faciale de pointe, tels que **DeepFace**, **FaceNet** et **ArcFace**, ont permis d'atteindre des jalons importants en termes de précision, même au niveau des performances humaines, et sont des références dans le domaine.

Si des gains substantiels sont réalisés avec l'augmentation de la complexité des conditions du monde réel, les occlusions ou la variabilité environnementale des visages masqués s'avèrent toujours difficiles. Toutes ces limitations indiquent la nécessité de continuer à améliorer les modèles et d'ajuster les méthodologies pour une fiabilité maximale compte tenu des conditions d'utilisation variables.

QUATRIÈME CHAPITRE – MÉTHODOLOGIE

4.1 INTRODUCTION

Ce chapitre expose la méthodologie adoptée pour mettre en œuvre notre système proposé pour la reconnaissance faciale, ainsi que la détection des visages masqués. Il se concentrera en outre sur le développement d'un système efficace et robuste pour identifier un individu, qu'il porte ou non un masque, avec une grande précision dans différents environnements. Il aborde également ce processus, qui comprend la préparation des données, le développement de modèles adaptés et leur validation par des tests approfondis. En fait, chaque étape, de la collecte des données à l'évaluation des performances du système, est importante pour obtenir des résultats fiables.

4.2 ENVIRONNEMENT DE TRAVAIL

L'environnement de travail est essentiel pour garantir des performances optimales lors de l'entraînement et de l'évaluation des modèles de détection et de reconnaissance faciale. Ce projet repose sur une combinaison de matériel performant et d'outils logiciels adaptés aux besoins du Deep Learning, en assurant une gestion fluide des bases de données et un environnement de développement flexible.

Matériel :

1. GPU (Graphics Processing Unit) :

Le GPU est un élément clé dans l'entraînement des modèles de Deep Learning, car il permet d'accélérer le traitement parallèle des calculs massifs requis par les réseaux de neurones convolutifs (CNN). Dans ce projet, nous avons utilisé un **GPU NVIDIA 3900**, qui est bien adapté aux tâches de calcul intensif nécessaires pour entraîner des modèles complexes avec des ensembles de données volumineux. Il offre une grande capacité de traitement pour optimiser l'efficacité et réduire le temps d'entraînement.

2. Processeur

Le processeur **Intel Core i7** a été utilisé pour gérer les tâches non parallélisées telles que

la gestion des fichiers, l'exécution des scripts Python, et l'intégration avec la base de données MySQL. Bien que le GPU soit essentiel pour l'entraînement des modèles, le CPU gère les tâches d'arrière-plan et permet un traitement fluide de l'ensemble du pipeline.

3. Mémoire

Une **mémoire vive de 32 Go** a été utilisée pour garantir une gestion fluide des opérations impliquant des données massives. La grande capacité de RAM permet de charger des images en haute résolution et de traiter les lots de données pendant l'entraînement des modèles, minimisant ainsi les goulots d'étranglement dus à la mémoire insuffisante.

Logiciel :

1. Python

Python est un langage de programmation largement utilisé dans le domaine du Deep Learning en raison de sa simplicité et de sa richesse en bibliothèques dédiées à l'intelligence artificielle. Python offre une flexibilité exceptionnelle pour le développement de modèles d'apprentissage automatique. Dans ce projet, Python a servi de base pour l'implémentation des algorithmes de reconnaissance faciale, l'entraînement des modèles, et la gestion des flux de données. Il est également intégré avec des frameworks comme TensorFlow et Keras pour faciliter la conception de réseaux neuronaux.

2. TensorFlow

TensorFlow est un framework open-source de Deep Learning développé par Google, qui permet de construire, entraîner et déployer des modèles d'apprentissage automatique. TensorFlow est particulièrement apprécié pour sa capacité à gérer des opérations à grande échelle et son intégration native avec le GPU, ce qui optimise l'entraînement des modèles de reconnaissance faciale. Dans ce projet, TensorFlow a été utilisé pour construire des réseaux de neurones convolutifs (CNN), modéliser la détection de visages et reconnaître des visages masqués en temps réel.

3. Keras

Keras est une interface de haut niveau qui fonctionne sur TensorFlow, facilitant la création et la gestion de modèles de Deep Learning. Keras est choisie pour sa simplicité et sa

capacité à rapidement prototyper des modèles complexes avec peu de code. Dans ce projet, Keras a été utilisée pour définir les architectures des modèles, en particulier la mise en place des couches convolutives et des couches de classification dans les réseaux de reconnaissance faciale et de détection de masques.

4. **MySQL**

MySQL est un système de gestion de bases de données relationnelles qui a été utilisé dans ce projet pour stocker les images et les résultats des modèles. Il permet de gérer efficacement les données d'entraînement, les étiquettes des visages (avec ou sans masque), et les vecteurs d'incorporation extraits des images. MySQL assure une structure organisée et offre des capacités de requête SQL pour accéder rapidement aux informations nécessaires à l'entraînement des modèles et à la validation des résultats.

5. **Windows**

Windows 11 a été utilisé comme système d'exploitation pour ce projet. Bien qu'Ubuntu soit souvent utilisé dans le domaine du Deep Learning, Windows 11 offre une compatibilité complète avec les bibliothèques Python, TensorFlow, et les drivers GPU NVIDIA. Il permet également une gestion aisée des environnements virtuels et une installation sans heurts des outils nécessaires à l'entraînement des modèles.

4.3 PRÉPARATION DE DONNÉES

La préparation des données est une phase cruciale pour garantir la robustesse, la précision, et la généralisation du modèle de détection de masques et de reconnaissance faciale. Une bonne qualité des données et leur traitement adéquat sont les clés pour entraîner un modèle capable de répondre aux défis rencontrés dans des conditions réelles. Dans le cadre de ce projet, deux bases de données principales ont été utilisées pour deux tâches distinctes : la détection de visages masqués et la reconnaissance de visages avec ou sans masques. Nous avons mis en place une méthodologie rigoureuse pour la collecte, la préparation, et l'augmentation des données afin de maximiser l'efficacité des modèles développés.

4.3.1 Base de données pour la détection de visages masqués : MaskedFace-Net

MaskedFace-Net [2] est l'une des bases de données de référence pour la détection de masques. Développée dans le contexte de la pandémie de COVID-19, cette base de données contient des milliers d'images de visages humains portant des masques de manière correcte, incorrecte, ou sans masque. Cette base de données a été sélectionnée pour son large éventail de visages dans des conditions variées (différents âges, ethnies, angles de prise de vue, et arrière-plans).

1. Description de la base de données MaskedFace-Net

- **Nombre d'images** : La base de données contient environ **140 000 images** de visages, dont plus de **70 000 images** de visages portant des masques. Les images sont divisées en trois catégories principales :
 - **Masques correctement portés** : Images où le masque couvre à la fois le nez et la bouche correctement.
 - **Masques mal portés** : Images où le masque ne couvre pas correctement le nez ou la bouche, ou où il est porté sous le menton.
 - **Sans masque** : Images de visages sans masque pour permettre la comparaison et le contraste avec les autres classes. (Voir figure 11)

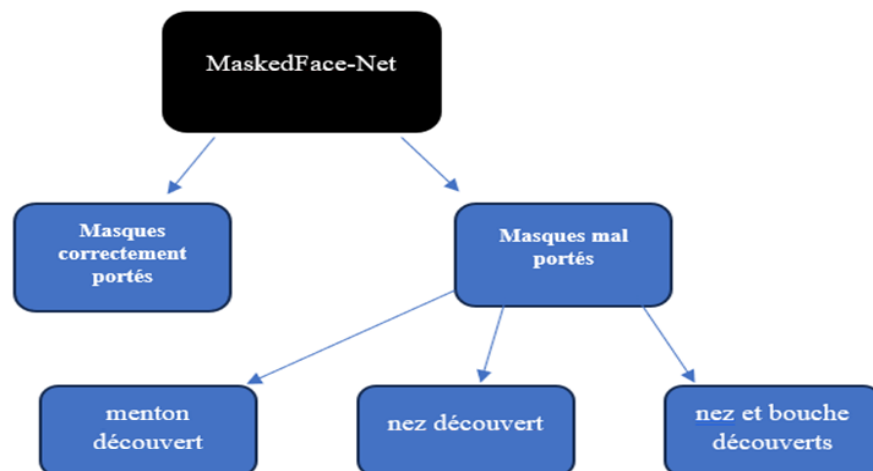


Figure 11 : Composition de la base de données MaskedFace-Net

- **Diversité des visages** : Les visages inclus dans la base de données proviennent de personnes de diverses origines ethniques et d'âges différents, ce qui rend le modèle plus robuste et évite les biais de reconnaissance faciale. De plus, les images sont capturées dans divers environnements, avec des conditions d'éclairage et d'arrière-plan très variables.

2. Pré-traitement des données

Avant d'entraîner les modèles, il est essentiel de pré-traiter les données pour garantir la compatibilité et la cohérence des entrées. Les étapes suivantes ont été suivies pour préparer les images de MaskedFace-Net :

- **Redimensionnement des images** : Toutes les images ont été redimensionnées à une taille standard de **224x224 pixels** afin d'assurer une compatibilité avec les modèles de réseaux de neurones convolutifs (CNN) utilisés. Cela permet également de réduire la complexité des calculs tout en maintenant un bon niveau de détail des caractéristiques faciales.
- **Normalisation** : Les valeurs de pixel des images ont été normalisées pour être comprises entre 0 et 1, facilitant ainsi l'entraînement du modèle en évitant les problèmes de gradients explosifs ou trop faibles. La normalisation aide également à rendre le modèle plus stable et plus rapide à converger.
- **Annotation** : Chaque image a été annotée avec une étiquette indiquant si le visage est masqué correctement, mal masqué, ou sans masque. Ces annotations sont cruciales pour que le modèle puisse apprendre à classer les visages dans ces trois catégories.

3. Augmentation des données

L'augmentation des données est une étape clé pour accroître la diversité du jeu de données et éviter le surapprentissage. Dans notre projet, plusieurs techniques d'augmentation des données ont été utilisées pour générer davantage de variations dans les images, simulant ainsi des conditions réelles variées. Cela permet au modèle d'être plus robuste et de généraliser à de nouveaux ensembles de données non vus.

- **Rotation** : Les images ont été tournées à des angles aléatoires (entre -20° et $+20^\circ$) afin de simuler différents angles de vue. Cela permet au modèle de mieux reconnaître les visages masqués même s'ils ne sont pas directement orientés face à la caméra.
- **Zoom et recadrage** : Des zooms aléatoires ont été appliqués aux images, permettant au modèle d'apprendre à reconnaître des visages à différentes distances et de s'adapter à des visages partiellement visibles.
- **Flip horizontal** : L'image a été retournée horizontalement dans certains cas. Cette technique est utile pour augmenter le nombre de variations possibles d'un même visage, en s'assurant que le modèle n'apprend pas uniquement sur des images statiques non inversées.
- **Modifications de la luminosité et du contraste** : Des variations aléatoires de la luminosité et du contraste ont été appliquées pour simuler différentes conditions d'éclairage (comme un faible éclairage ou une lumière très vive), ce qui rend le modèle plus résistant aux variations d'éclairage en situation réelle.

4. Création des ensembles de formation, validation et test

Pour évaluer la performance du modèle, le jeu de données MaskedFace-Net a été divisé en trois sous-ensembles :

- **Ensemble de formation (70%)** : Cet ensemble est utilisé pour entraîner le modèle. Il contient la majorité des images, couvrant toutes les classes (masqué, mal masqué, sans masque).
- **Ensemble de validation (15%)** : Pendant l'entraînement, cet ensemble est utilisé pour surveiller les performances du modèle et éviter le surapprentissage. Les performances sur cet ensemble permettent de décider quand arrêter l'entraînement (early stopping).
- **Ensemble de test (15%)** : Ce sous-ensemble est utilisé uniquement après l'entraînement pour évaluer la capacité du modèle à généraliser sur des données qu'il n'a jamais vues.

4.3.2 Base de données pour la reconnaissance de visages masqués : Million Celebs

La reconnaissance de visages avec et sans masque nécessite un ensemble de données riche et varié. Pour cela, nous avons utilisé la base de données **Million Celebs** [3], qui contient des millions d'images de visages de célébrités, capturées sous différents angles et avec différentes expressions faciales. Cette base de données est particulièrement bien adaptée pour la tâche de reconnaissance faciale, car elle couvre un large éventail de conditions réalistes.

1. Description de la base de données Million Celebs

- **Nombre d'images** : **Million Celebs** contient environ **1 million d'images** de visages humains, offrant une grande variété d'expressions faciales, d'angles de vue, d'âges, et d'ethnies. Cela en fait une base de données idéale pour entraîner un modèle de reconnaissance faciale capable de généraliser à des environnements réels. (Voir figure 12)

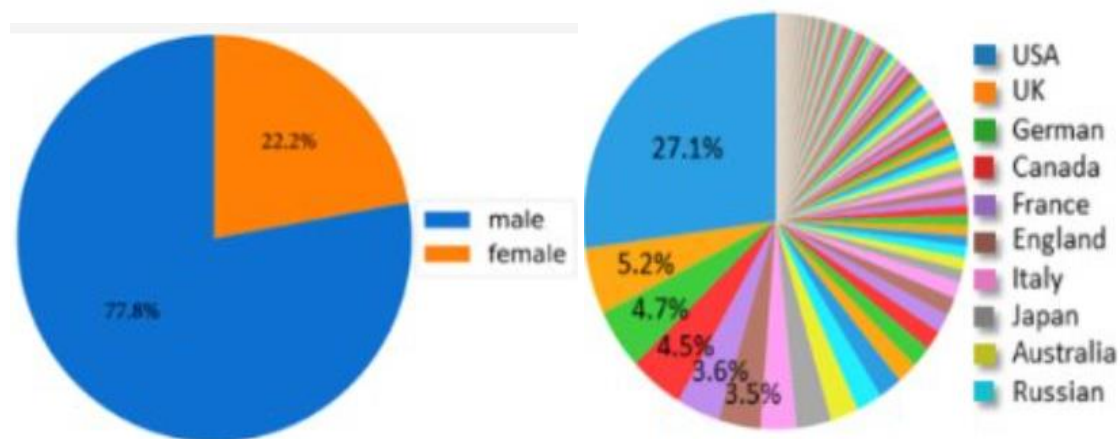


Figure 12 : Statistiques d'âge et nationalité de la base de données (Million Celebs) [3]

- **Masques ajoutés synthétiquement** : Pour rendre cette base de données applicable à la reconnaissance de visages masqués, des masques faciaux ont été ajoutés synthétiquement à certaines images. Ces masques ont été appliqués à des images aléatoires de la base de données, créant ainsi des versions avec et sans masque de chaque visage.

2. Préparation des données

- **Alignement des visages** : Avant d'ajouter les masques synthétiques, les visages ont été alignés en utilisant des algorithmes de détection de points clés (landmarks). Cela garantit que les masques sont placés correctement sur le nez et la bouche des individus dans les images. Cet alignement est crucial pour améliorer la précision du modèle de reconnaissance.
- **Redimensionnement des images** : Comme pour MaskedFace-Net, toutes les images de Million Celebs ont été redimensionnées à **224x224 pixels** pour être compatibles avec les modèles de reconnaissance faciale basés sur les réseaux de neurones convolutifs.
- **Normalisation des images** : Les valeurs des pixels ont été normalisées (entre 0 et 1) pour uniformiser les données et faciliter l'entraînement du modèle de reconnaissance faciale.

3. Création des ensembles de formation, validation et test

Comme pour la détection de masques, la base de données Million Celebs a été divisée en trois sous-ensembles :

- **Ensemble de formation (70%)** : Cet ensemble est utilisé pour entraîner le modèle de reconnaissance faciale. Il inclut des visages avec et sans masques pour chaque célébrité.
- **Ensemble de validation (15%)** : Pendant l'entraînement, cet ensemble permet de surveiller la performance du modèle et de déterminer quand arrêter l'entraînement pour éviter le surapprentissage.
- **Ensemble de test (15%)** : Cet ensemble est réservé pour l'évaluation finale du modèle. Il permet de vérifier si le modèle est capable de reconnaître des visages qu'il n'a jamais vus, en particulier dans des conditions où un visage est partiellement masqué.

4.4 DÉTECTION DE VISAGES

La détection des visages représente la première étape de tout système de reconnaissance faciale. Il s'agit de localiser les visages présents dans une image ou une vidéo avant de pouvoir appliquer les techniques de reconnaissance ou d'utiliser les algorithmes de détection de masque. Cette étape préliminaire est très importante pour garantir l'efficacité et la précision des modèles de reconnaissance faciale. Les techniques de détection des visages qui ont été utilisées dans ce travail incluent celles basées à la fois sur des approches traditionnelles et sur des modèles basés sur le Deep Learning.

Approches traditionnelles : Haar Cascade

L'une des méthodes les plus populaires et rapides pour la détection de visages est l'algorithme **Haar Cascade**, introduit par **Viola et Jones (2001)** [26]. Cet algorithme repose sur l'utilisation de caractéristiques de Haar pour détecter les visages en temps réel. Il est couramment utilisé dans des systèmes de reconnaissance faciale pour des applications légères où les ressources en calcul sont limitées.

Approches basées sur le Deep Learning : SSD (Single Shot Detector)

- Pour améliorer la précision de la détection des visages dans des environnements complexes, nous avons adopté un modèle basé sur SSD. Le SSD peut exécuter une détection d'objets par apprentissage profond en un seul passage à des vitesses élevées avec une grande précision, ce qui inclut la détection des visages.

1. Comment fonctionne le SSD

- Dans le modèle SSD, une architecture convolutionnelle est mise en œuvre qui divise l'image en une grille et prédit des boîtes englobantes pour chaque région de la grille. Chacune des boîtes correspond à une certaine probabilité d'appartenir à la classe spécifique considérée, par exemple, un visage. Contrairement aux méthodes classiques, comme Haar Cascade, le SSD est capable de détecter les visages sous différents angles et dans diverses conditions d'éclairage.

- **Grille d'ancrage** : L'image est divisée en une grille de cellules, et chaque cellule est responsable de prédire plusieurs boîtes englobantes avec des échelles et des rapports d'aspect différents. Cette approche permet de détecter des visages de différentes tailles, même dans des images de résolutions variées [40].
- **Réseau convolutif** : SSD utilise des **convolutions** pour extraire des caractéristiques à différents niveaux de granularité, permettant ainsi de détecter des visages dans des zones d'images complexes ou peu contrastées. Les **convolutions profondes** aident à extraire des motifs complexes tout en réduisant le nombre de paramètres du modèle, ce qui améliore la vitesse de traitement.

2. Utilisation de SSD avec Caffe

Pour ce projet, nous avons utilisé un modèle SSD pré-entraîné sur la base de données **ResNet10** avec l'architecture **Caffe**. Ce modèle est chargé à l'aide des fichiers **deploy.prototxt.txt** et **res10_300x300_ssd_iter_140000_fp16.caffemodel**, qui contiennent la configuration et les poids du réseau [34]. Ces fichiers permettent de charger le modèle dans un environnement Python et de l'utiliser pour la détection en temps réel.

3. Avantages et limites de SSD

- **Avantages** :
 - **Détection rapide et précise** : SSD offre une détection rapide en une seule passe, ce qui en fait un modèle idéal pour des applications en temps réel, telles que la surveillance ou l'identification dans des vidéos [40].
 - **Robustesse** : Le modèle est capable de détecter des visages sous différents angles et dans des environnements avec des conditions d'éclairage variées. Il est également plus robuste pour les visages partiellement masqués par rapport aux approches traditionnelles.
 - **Adaptabilité aux visages masqués** : Contrairement à Haar Cascade, SSD peut encore détecter des visages même si une partie du visage est masquée, car il repose sur l'extraction de caractéristiques à plusieurs niveaux [34].

- **Limites :**

- **Coût computationnel :** Bien que rapide, SSD nécessite une puissance de calcul plus importante que Haar Cascade, notamment lors de l'entraînement. Il est donc préférable de l'utiliser sur des systèmes équipés de GPU pour un traitement optimal [40].
- **Sensibilité aux visages très petits :** SSD peut parfois rencontrer des difficultés à détecter des visages très petits ou à longue distance dans une image [34].

Fusion des modèles pour une détection optimisée

Dans ce projet, nous avons choisi de combiner les approches **Haar Cascade** et **SSD** pour profiter des avantages de chacune. L'idée est d'utiliser Haar Cascade pour des scénarios simples, où les visages sont principalement frontaux et sans obstacles, et d'utiliser SSD pour des scénarios plus complexes, avec des angles variés ou des masques faciaux.

1. Processus de fusion

- **Étape 1 :** Lorsqu'une image est capturée, elle est d'abord analysée par Haar Cascade, qui effectue une détection rapide des visages frontaux [26].
- **Étape 2 :** Si Haar Cascade ne parvient pas à détecter un visage ou si l'image contient des visages sous différents angles, SSD est utilisé pour affiner la détection et identifier les visages qui n'ont pas été détectés par Haar Cascade [34].

2. Avantages de la fusion

Ce système hybride maximise l'approche de détection des visages en tirant parti des avantages de la vitesse de Haar Cascade et de la précision/robustesse de SSD. Ainsi, la gamme de scénarios environnementaux opérationnels que ce système peut couvrir va du simple, par exemple, la détection de visages bien alignés sur des photos, aux paramètres de scénarios

relativement complexes où il existe des poses inclinées et des conditions de masquage dans le cas de la surveillance.

Évaluation de la performance des modèles de détection

La performance des modèles de détection a été évaluée en termes de **précision**, **rappel**, et **F1-score**. Ces métriques permettent de mesurer la capacité des modèles à détecter correctement les visages dans différentes conditions.

- **Précision** : Mesure la proportion de détections correctes parmi toutes les détections faites par le modèle [40].
- **Rappel** : Mesure la proportion de visages correctement détectés par rapport au nombre total de visages dans l'image [26].
- **F1-score** : La moyenne harmonique entre la précision et le rappel, offrant une évaluation globale de la performance du modèle [40].

Les résultats montrent que la combinaison de Haar Cascade et SSD permet d'obtenir une performance globale plus robuste, en particulier pour des images contenant des visages masqués ou dans des angles difficiles.

4.5 EXTRACTION DES ASPECTS DISTINCTIFS

L'extraction de caractéristiques distinctives du visage est un processus important dans la reconnaissance faciale et la détection de masques. Le processus transforme une image brute en un ensemble de caractéristiques très uniques qui peuvent ensuite être utilisées pour distinguer les individus ou détecter les masques. Ces vecteurs, appelés vecteurs d'intégration, ou caractéristiques, sont générés par les modèles **CNN** qui apprennent à extraire des informations pertinentes telles que la forme du visage, la position des yeux, la structure du nez ou les motifs de la peau. Dans le cadre de cette recherche, la combinaison de modèles utilisée pour l'extraction de caractéristiques dans les images comprenait un modèle **CNN** personnalisé et un modèle **MobileNetV2**. Ces deux architectures capturent ensemble les

aspects locaux et globaux, indiquant respectivement les détails fins et les caractéristiques d'ensemble des visages.

Modèle CNN personnalisé pour l'extraction de caractéristiques locales

Un modèle **CNN personnalisé** a été conçu pour extraire des caractéristiques locales des visages dans les images. Ces caractéristiques locales se réfèrent aux détails fins d'une image, tels que les bords, les textures, ou les motifs qui sont spécifiques à une région du visage.

1. Architecture du modèle CNN

Le modèle CNN personnalisé utilisé dans cette étude comporte plusieurs couches convolutives et de **max-pooling**, qui extraient progressivement des caractéristiques plus abstraites de l'image.

- **Couches convolutives** : Ces couches sont responsables de l'extraction des caractéristiques locales à partir de l'image d'entrée. Chaque filtre dans une couche convolutive analyse une petite région de l'image et détecte des motifs comme les contours des yeux, du nez, ou des lèvres.
- **Max-Pooling** : Après chaque couche convolutive, une couche de **Max-Pooling** est utilisée pour réduire la taille spatiale des caractéristiques tout en préservant les informations les plus importantes. Le pooling permet de réduire la complexité du modèle et d'augmenter sa capacité de généralisation.
- **Flatten** : Une fois que les caractéristiques ont été extraites et réduites, la couche **Flatten** transforme les matrices de caractéristiques en un vecteur plat qui est ensuite passé à travers une couche entièrement connectée pour la classification ou la reconnaissance faciale.

2. Avantages du modèle CNN personnalisé

- **Spécificité locale** : Le modèle CNN personnalisé est capable de capturer des détails fins du visage, ce qui est particulièrement utile pour la reconnaissance faciale. Les caractéristiques locales, telles que les contours des yeux, la forme du nez, ou la texture de la peau, jouent un rôle important dans la différenciation des individus [10].
- **Efficacité** : Grâce à sa simplicité et à sa légèreté, le modèle CNN personnalisé peut être entraîné relativement rapidement, tout en offrant une bonne capacité à distinguer des visages similaires.

3. Limites du modèle CNN personnalisé

- **Sensibilité aux variations globales** : Bien que performant pour extraire des caractéristiques locales, un CNN personnalisé peut être limité pour capturer des informations globales, comme la structure d'ensemble du visage ou l'orientation du visage. Cela peut affecter la reconnaissance faciale dans des scénarios où le visage est masqué ou partiellement visible.

MobileNetV2 pour l'extraction de caractéristiques globales

Pour compléter les caractéristiques locales extraites par le modèle CNN personnalisé, nous avons utilisé un modèle **MobileNetV2**, qui est une architecture de réseau de neurones convolutifs légère et optimisée pour l'extraction de caractéristiques globales à partir d'images. MobileNetV2 est particulièrement adapté aux environnements où la puissance de calcul est limitée, tout en maintenant une haute précision pour des tâches de classification et de reconnaissance [41].

1. Fonctionnement de MobileNetV2

MobileNetV2 utilise des **couches convolutives à profondeur réduite (depthwise separable convolutions)**, qui permettent de réduire considérablement le nombre de paramètres et les besoins en calcul par rapport aux réseaux CNN traditionnels. En utilisant cette approche, MobileNetV2 peut capturer des informations globales importantes tout en restant léger et rapide.

- **Couches convolutives à profondeur réduite** : Contrairement aux convolutions standards qui traitent toutes les dimensions des filtres en une seule opération, MobileNetV2 décompose cette opération en deux étapes, ce qui permet de réduire le coût computationnel tout en conservant une bonne performance [41].
- **GlobalAveragePooling2D** : Une couche de **GlobalAveragePooling2D** est utilisée pour réduire davantage la dimensionnalité des caractéristiques globales extraites. Cette couche moyenne les caractéristiques spatiales de chaque carte d'activation, générant ainsi une représentation compacte et robuste du visage complet.

2. Avantages de MobileNetV2

- **Légèreté et rapidité** : MobileNetV2 est conçu pour être extrêmement léger et rapide, tout en offrant des performances comparables à des architectures CNN plus grandes. Cela le rend idéal pour des applications de reconnaissance faciale en temps réel [42].
- **Robustesse aux variations globales** : Grâce à ses caractéristiques globales, MobileNetV2 est capable de capturer des informations d'ensemble du visage, même lorsque certaines parties sont masquées ou obstruées.

3. Limites de MobileNetV2

- **Moins de détails fins** : MobileNetV2 se concentre principalement sur l'extraction de caractéristiques globales, ce qui peut limiter sa capacité à capturer des détails fins d'une image. C'est pourquoi il est utilisé en complément du CNN personnalisé pour une extraction plus complète des caractéristiques.

Fusion des caractéristiques locales et globales

Pour tirer parti des deux approches, nous avons fusionné les caractéristiques extraites par le modèle CNN personnalisé et par MobileNetV2. Cette fusion permet d'obtenir une représentation complète du visage, capturant à la fois les détails fins et les informations globales.

1. Processus de fusion

- **Combinaison des vecteurs de caractéristiques** : Les caractéristiques extraites par le CNN personnalisé (détails locaux) et MobileNetV2 (caractéristiques globales) sont concaténées pour former un vecteur de caractéristiques fusionné. Ce vecteur combine à la fois les informations fines sur les contours et les textures du visage, et les informations globales sur la structure du visage.
- **Passage dans une couche Dense** : Une fois les caractéristiques fusionnées, elles sont passées dans une couche entièrement connectée (**Dense Layer**) qui apprend à utiliser ces caractéristiques combinées pour effectuer la classification ou la reconnaissance faciale.

2. Avantages de la fusion

- **Meilleure généralisation** : En combinant des caractéristiques locales et globales, le modèle est capable de mieux généraliser à des visages sous différents angles ou partiellement obstrués. Cela améliore considérablement la précision dans des scénarios complexes, tels que la reconnaissance de visages masqués [43].
- **Robustesse face aux masques** : La combinaison des caractéristiques locales (yeux, front) et globales (forme générale du visage) permet au modèle de reconnaître efficacement des visages même lorsqu'une partie est cachée par un masque.

Evaluation des performances de l'extraction de caractéristiques

La performance des modèles d'extraction de caractéristiques a été évaluée à l'aide de plusieurs métriques, notamment la **précision**, le **rappel**, et le **F1-score**. Ces métriques ont été utilisées pour mesurer la capacité du modèle à extraire des caractéristiques pertinentes et à différencier efficacement les individus dans différents contextes.

- **Précision** : Mesure la capacité du modèle à prédire correctement l'identité d'un visage ou la présence d'un masque à partir des caractéristiques extraites [43].

- **Rappel** : Mesure la proportion de visages correctement identifiés parmi ceux présents dans l'image. Cela permet de savoir si le modèle extrait les caractéristiques importantes [10].
- **F1-score** : La moyenne harmonique entre la précision et le rappel, utilisée pour évaluer l'équilibre global du modèle.

Les résultats montrent que la fusion des caractéristiques locales et globales permet d'obtenir une meilleure performance dans la reconnaissance des visages, en particulier dans des conditions où une partie du visage est masquée.

4.6 CLASSIFICATION DES VISAGES

La classification des visages est l'une des étapes majeures qui suivent l'extraction de caractéristiques distinctes d'un visage. Après l'extraction des vecteurs de caractéristiques à l'aide des **modèles CNN** et **MobileNetV2**, il est temps de les utiliser pour attribuer des identités ou des étiquettes aux visages présents dans les images. Il est important de noter que ce type de classification peut également classer la distinction entre les visages portant un masque, un masque mal porté ou aucun masque afin d'assurer une reconnaissance appropriée dans différents scénarios. Pour notre travail, la classification a été effectuée via un modèle DNN, qui utilisait comme entrée les caractéristiques extraites et les mappait dans une prédiction de classe. Il se compose de plusieurs couches entièrement connectées et d'une couche de sortie avec la fonction d'activation **softmax**, car il s'agit d'un problème de classification multi-classe.

Description du processus de classification

1. Entrée des caractéristiques extraites

Celles-ci sont ensuite combinées dans un vecteur avec les caractéristiques extraites par le modèle CNN personnalisé concernant les caractéristiques locales et globales, ainsi que celles de **MobileNetV2**, en passant par plusieurs couches de réseau dense. Ces vecteurs de caractéristiques englobent la plupart des informations des visages, qu'il s'agisse de la structure des yeux, de la forme du nez ou de la texture de la peau.

2. Couches denses et fonction d'activation

Une fois que les vecteurs de caractéristiques sont créés, ils sont envoyés dans des couches entièrement connectées (ou **Dense Layers**), qui servent à établir des connexions entre toutes les unités du réseau. Ces couches permettent au modèle d'apprendre des combinaisons non linéaires de caractéristiques extraites et de mieux distinguer les différentes classes (visages masqués ou non masqués).

- **Couches Dense** : Chaque couche dense est composée d'un ensemble de neurones entièrement connectés. Dans notre modèle, une première couche dense de **512 neurones** est utilisée pour traiter les caractéristiques combinées, suivie d'une couche **Dropout** avec un taux de 0,5 pour éviter le surapprentissage.
- **Fonction d'activation ReLU** : La fonction d'activation **ReLU (Rectified Linear Unit)** est utilisée dans les couches intermédiaires pour introduire de la non-linéarité dans le modèle. Cette non-linéarité permet de capturer des relations complexes entre les caractéristiques extraites du visage [44].

3. Couche de sortie avec Softmax

La dernière couche du modèle de classification est une couche dense avec **3 neurones**, correspondant aux trois classes cibles de cette étude :

- **Visage avec masque,**
- **Visage sans masque,**
- **Visage avec masque mal porté.**

Cette couche utilise la fonction d'activation **Softmax**, qui génère une distribution de probabilité pour chaque classe. La classe ayant la plus haute probabilité est sélectionnée comme la prédiction finale.

- **Softmax** : La fonction **Softmax** est particulièrement adaptée à la classification multiclasse, car elle transforme les sorties du modèle en probabilités normalisées qui indiquent la

probabilité que le visage appartienne à chaque classe [45]. Cela permet au modèle de produire une seule classe de sortie avec une forte confiance.

Méthodes de régularisation

L'un des défis dans l'entraînement de modèles de classification est le risque de **surapprentissage** (overfitting), c'est-à-dire que le modèle s'adapte trop aux données d'entraînement et performe mal sur de nouvelles données. Pour éviter cela, plusieurs techniques de régularisation ont été utilisées.

1. Dropout

La **technique de Dropout** a été appliquée dans les couches entièrement connectées du modèle pour éviter le surapprentissage. Dropout consiste à désactiver de manière aléatoire un certain pourcentage de neurones pendant l'entraînement, forçant ainsi le réseau à apprendre des représentations plus générales plutôt que de s'appuyer sur certaines connexions spécifiques.

- **Dropout avec un taux de 0,5** : Dans notre modèle, un taux de Dropout de **0,5** a été utilisé, ce qui signifie que 50 % des neurones de la couche dense sont désactivés à chaque itération d'entraînement. Cette technique aide à améliorer la robustesse du modèle et à éviter qu'il mémorise les données d'entraînement [46].

2. Early Stopping

L'**Early Stopping** est une technique de régularisation qui arrête l'entraînement du modèle lorsque les performances sur l'ensemble de validation cessent de s'améliorer après un certain nombre d'époques. Cela évite que le modèle continue à s'entraîner après avoir atteint son meilleur point de généralisation.

- **Critère d'arrêt** : Un critère d'**Early Stopping** a été utilisé avec un délai de **10 époques**, ce qui signifie que si le modèle ne montre pas d'amélioration de la précision sur l'ensemble de validation pendant 10 époques consécutives, l'entraînement est automatiquement arrêté.

Classification des visages masqués et sans masque

Cette classification des visages en trois classes - visage masqué, visage mal masqué, visage sans masque - est particulièrement importante pour la présente étude, car elle permet d'estimer la variation des résultats provoquée par les masques lors de la reconnaissance des visages. Le modèle doit être capable de distinguer non seulement les visages, mais également de détecter l'absence ou la mauvaise position du masque.

1. Précision de la classification

La précision de la classification a été mesurée pour les trois classes mentionnées, avec un accent particulier sur la capacité du modèle à reconnaître les visages partiellement obstrués par des masques.

- **Taux de précision** : Le modèle a montré un taux de précision supérieur à **93 %** pour la détection des visages correctement masqués et non masqués. La reconnaissance des visages avec un masque mal porté a un taux de précision légèrement inférieur, autour de **89 %**, en raison de la variabilité des masques mal positionnés [42].

2. Importance de l'extraction des caractéristiques

La puissance du modèle dépend donc en grande partie de l'efficacité de l'extraction des caractéristiques par le **modèle CNN** et **MobileNetV2**. Il capture à la fois des informations locales, telles que les yeux et le nez, et des informations globales sur la structure du visage. Par conséquent, il aurait une meilleure distinction entre les visages masqués et non masqués au cas où le masque occulte une partie du visage [41].

Évaluation des performances de classification

Les performances du modèle de classification ont été évaluées à l'aide de plusieurs métriques, notamment :

1. Précision (Accuracy)

La **précision** mesure le pourcentage total de prédictions correctes faites par le modèle. Elle est calculée comme le rapport entre le nombre de prédictions correctes et le nombre total de prédictions effectuées. Dans notre cas, cela inclut la reconnaissance correcte des visages masqués, mal masqués, et sans masques [45].

2. Rappel (Recall)

Le **rappel** (ou sensibilité) mesure la capacité du modèle à identifier correctement toutes les instances d'une classe donnée. Dans ce projet, le rappel est utilisé pour vérifier si le modèle détecte correctement tous les visages masqués ou mal masqués dans les images test.

3. F1-Score

Le **score F1** est la mesure qui associe précision et rappel. Il exprime la moyenne harmonique des deux métriques ; il est donc très utile lorsque nous souhaitons considérer le compromis entre ces deux aspects. Dans notre cas, le score F1 est pertinent car il garantira que le modèle est non seulement précis mais qu'il capturera également tout ce qui concerne les visages masqués ou mal masqués.

Comparaison avec d'autres méthodes de classification

Pour évaluer la robustesse de notre modèle de classification, nous avons comparé les résultats obtenus avec ceux de modèles plus simples, tels que des **machines à vecteurs de support (SVM)** et des **forêts aléatoires**. Bien que ces méthodes aient montré des performances respectables, elles se sont révélées moins efficaces pour gérer des visages masqués et partiellement obstrués que le modèle DNN utilisé dans notre étude.

4.7 IDENTIFICATION

L'identification des visages est l'un des processus les plus importants et les plus complexes d'un système de reconnaissance faciale. Contrairement à la notion de classification qui attribue un visage soit à la catégorie avec masque, soit à la catégorie sans masque, l'identification des visages associe un visage détecté à une seule identité particulière issue d'une base de données de personnes. Cela nécessite non seulement l'extraction précise des caractéristiques faciales, mais aussi l'efficacité de la comparaison des visages détectés avec ceux déjà enregistrés. À cet égard, l'identification repose sur des techniques sophistiquées impliquant l'extraction et la comparaison de **vecteurs d'intégration** à partir de modèles tels que **MobileNetV2** et **FaceNet**. Ces vecteurs contiendraient des informations distinctives sur un visage pour ensuite effectuer des comparaisons de visages - précises, bien sûr - même lorsque ces visages sont partiellement masqués par des masques.

Processus d'identification des visages

Le processus d'identification des visages suit plusieurs étapes clés, allant de l'extraction des caractéristiques à la comparaison des vecteurs d'incorporation avec une base de données d'individus connus.

1. Extraction des vecteurs d'incorporation (embeddings)

Le processus d'identification commence par l'extraction de **vecteurs d'intégration** à partir des visages détectés. Il s'agit de représentations numériques compactes de visages provenant de modèles **MobileNetV2** ou **FaceNet**. Ces modèles peuvent extraire des informations faciales pertinentes tout en réduisant la dimensionnalité des données pour des comparaisons efficaces.

- **MobileNetV2** : Ce modèle est utilisé pour extraire des caractéristiques globales et des détails du visage, même lorsque celui-ci est masqué. MobileNetV2 est particulièrement utile pour obtenir des embeddings robustes grâce à ses couches convolutives à profondeur réduite [41].

- **FaceNet** : FaceNet génère des vecteurs d'incorporation de 128 dimensions qui capturent les traits distinctifs du visage [24]. Ce modèle a été choisi pour sa précision dans les tâches de reconnaissance faciale, même en présence de variations d'angle, d'éclairage, ou d'occlusion partielle (comme des masques).

2. Comparaison des vecteurs d'incorporation

Une fois les vecteurs d'incorporation extraits, l'identification consiste à les comparer avec ceux déjà stockés dans une base de données d'individus enregistrés. La comparaison est basée sur des mesures de distance, qui permettent de quantifier la similarité entre les vecteurs du visage détecté et ceux de la base de données.

- **Distance euclidienne** : La **distance euclidienne** est utilisée comme mesure de similarité. Elle calcule la distance entre deux vecteurs d'incorporation dans un espace multidimensionnel. Plus la distance est faible, plus les visages comparés sont similaires. Si la distance est inférieure à un seuil défini, le visage est reconnu comme correspondant à un individu de la base de données [24].

$$d(p, q) = \sqrt{\sum_{i=1}^n (p_i - q_i)^2}$$

Où p et q représentent les vecteurs d'incorporation du visage détecté et du visage de la base de données.

3. Identification des visages masqués

Le défi de l'identification se pose lorsque certains traits du visage sont occultés, mais grâce à la concentration principale des modèles dans ce travail sur les yeux et le front, ils ont effectivement réussi à extraire des informations distinctives et à permettre une identification fiable.

- **Approche basée sur la partie supérieure du visage** : Lorsque le visage est masqué, l'algorithme se concentre principalement sur les yeux, les sourcils, et la forme du front pour générer des vecteurs d'incorporation. Ces caractéristiques suffisent généralement pour différencier les individus, car elles sont moins affectées par la présence d'un masque.
- **Réduction des biais d'identification** : Les modèles ont été entraînés à reconnaître des visages avec et sans masques afin de minimiser les erreurs d'identification dues à l'obstruction faciale partielle. Cela permet de réduire les biais algorithmiques qui pourraient favoriser la reconnaissance des visages non masqués par rapport aux visages masqués.

Méthodes de gestion des erreurs d'identification

L'un des défis majeurs dans l'identification faciale est de gérer les **faux positifs** (lorsqu'un visage est incorrectement identifié) et les **faux négatifs** (lorsqu'un visage ne parvient pas à être reconnu malgré sa présence dans la base de données). Plusieurs stratégies ont été mises en œuvre pour limiter ces erreurs.

1. Ajustement des seuils de décision

Le seuil utilisé pour la **distance euclidienne** est essentiel pour définir si un visage est reconnu ou non. Un seuil trop bas pourrait entraîner un grand nombre de faux négatifs, tandis qu'un seuil trop élevé pourrait provoquer des faux positifs. L'ajustement de ce seuil a été effectué de manière empirique en fonction des performances du modèle sur l'ensemble de validation.

- **Optimisation du seuil** : Un processus d'optimisation a été utilisé pour trouver le seuil de distance euclidienne qui maximise la précision de l'identification tout en minimisant les erreurs.

2. Gestion des visages non identifiés

Dans certains cas, des visages détectés ne correspondent à aucun vecteur d'incorporation dans la base de données, soit parce qu'ils n'ont jamais été enregistrés, soit en raison d'une obstruction excessive (par exemple, un masque couvrant la majorité du visage). Dans ces cas, des stratégies d'identification non résolues ont été mises en place.

- **Indication d'inconnu** : Lorsque la distance euclidienne entre le vecteur du visage détecté et tous les vecteurs dans la base de données dépasse un seuil élevé, le visage est marqué comme **inconnu**, indiquant qu'il ne correspond à aucun individu enregistré.

3. Réévaluation des caractéristiques après l'identification

Dans le cadre de systèmes de reconnaissance multi-couches, la réévaluation des caractéristiques extraites peut être nécessaire lorsque le modèle hésite entre plusieurs identités potentielles pour un visage donné. Dans ce cas, le processus d'extraction et de comparaison est réévalué pour garantir une identification plus précise.

Entraînement du modèle d'identification

Le modèle d'identification a été entraîné à l'aide de la base de données **Million Celebs**, qui contient des millions d'images de visages sous différents angles, avec et sans masques. Des masques ont été ajoutés synthétiquement à certains visages pour permettre au modèle de s'adapter aux scénarios de visages partiellement obstrués.

- **Augmentation des données** : Afin de rendre le modèle plus robuste, des techniques d'augmentation des données ont été utilisées, telles que la rotation, le zoom, et les ajustements d'éclairage, pour exposer le modèle à une grande variété de situations [10].
- **Validation croisée** : Le modèle a été validé par une approche de **validation croisée**, divisant les données en plusieurs sous-ensembles pour garantir que l'identification est robuste même sur des ensembles de données non vus.

Limitations de l'identification des visages masqués

L'identification des visages masqués présente certaines limites en raison de la perte d'informations faciales. Bien que les parties visibles du visage suffisent dans de nombreux cas pour une reconnaissance précise, certains défis demeurent :

- **Occlusion excessive** : Lorsque le masque couvre une part trop importante du visage (par exemple, le nez, la bouche, et une partie des joues), la précision de l'identification diminue, car les caractéristiques visibles ne suffisent plus à distinguer l'individu.

- **Variabilité des masques** : La forme, la couleur, et la texture des masques peuvent également affecter la performance d'identification, notamment lorsque les masques couvrent une partie des yeux ou du front.

4.8 ARCHITECTURE DE NOTRE SYSTÈME

L'architecture signifie essentiellement la détection et la reconnaissance faciale avec la gestion des masques, le fondement même du projet. Elle repose sur l'intégration d'un certain nombre de techniques avancées de **Deep Learning** et de Computer Vision : il y a l'intégration de modèles pré-entraînés en plus des réseaux personnalisés lors du traitement et de l'analyse des images. Il s'agira donc d'une architecture qui devra faire une prévision pour plusieurs tâches : la détection des visages, la classification de l'état du masque - avec, sans ou mal porté - et enfin identifier la personne portant un masque ou non.

Elle comprend une variété de modules interconnectés, qui exécutent un certain pipeline bien défini. Chacun de ces modules joue un rôle particulier dans le processus général de détection, de classification et d'identification.

4.8.1 Architecture du système de détection de masque

L'architecture système de détection de masque est basée sur un modèle combiné comprenant un CNN personnalisé et MobileNetV2. Elle tente de classer si un visage porte correctement son masque, ne le porte pas ou s'il est mal porté. Ainsi, les différentes étapes de l'architecture sont les suivantes :

1. Prétraitement des images

- **Entrée** : Les images de visages sont capturées ou importées.
- **Redimensionnement** : Chaque image est redimensionnée à **224x224 pixels**.
- **Normalisation** : Les valeurs des pixels sont normalisées pour être comprises entre 0 et 1, assurant une cohérence pour l'entraînement du modèle.

2. Détection des visages

- **Modèle SSD (Single Shot Detector)** : Ce modèle est utilisé pour détecter les visages dans l'image, même si ceux-ci sont partiellement masqués ou orientés sous différents angles. Les coordonnées des visages détectés sont renvoyées pour les étapes suivantes.

3. Extraction des caractéristiques (Modèle combiné)

- **CNN personnalisé** : Ce réseau de neurones convolutifs personnalisé extrait les caractéristiques locales des visages détectés, telles que les contours et les détails spécifiques au masque (par exemple, les contours des yeux et du masque).
- **MobileNetV2** : MobileNetV2 est utilisé pour extraire des caractéristiques globales des visages, en capturant la forme générale du visage, la position des yeux, et la partie supérieure du masque.

4. Fusion des caractéristiques

- Les caractéristiques extraites par les deux modèles sont fusionnées pour obtenir une représentation complète du visage, combinant à la fois des informations locales (détails du masque) et globales (structure du visage).

5. Classification du masque

- **Couche Dense** : Une couche entièrement connectée reçoit les caractéristiques fusionnées pour effectuer la classification.
- **Softmax** : La fonction d'activation Softmax est utilisée pour classifier les visages détectés dans trois catégories :
 - **Masque correctement porté,**
 - **Sans masque,**
 - **Masque mal porté.**
 -

6. Sortie

- Le système renvoie l'état du masque pour chaque visage détecté.

4.8.2 Architecture du système de reconnaissance faciale

Le système de reconnaissance faciale utilise une architecture similaire mais est optimisé pour l'identification d'individus, qu'ils portent ou non un masque. L'objectif est de comparer les visages détectés avec ceux d'une base de données pour les identifier.

1. Prétraitement des images

- **Entrée** : Les images ou vidéos contenant des visages sont capturées ou importées.
- **Redimensionnement** : Chaque image est redimensionnée à **224x224 pixels**.
- **Normalisation** : Les valeurs des pixels sont normalisées pour assurer une compatibilité avec le modèle.

2. Détection des visages

Une fois les images prétraitées, le système passe à la détection des visages. Cette étape consiste à localiser les visages présents dans les images ou les vidéos, qu'ils soient masqués ou non. Deux méthodes complémentaires sont utilisées pour cette tâche : le modèle Haar Cascade et le modèle SSD (Single Shot Detector).

- **Haar Cascade** : Utilisé pour sa rapidité et sa capacité à détecter efficacement les visages frontaux dans des images simples [26].
- **SSD avec Caffé** : Le modèle SSD (Single Shot Detector) est utilisé pour des scénarios plus complexes où les visages sont partiellement obstrués ou présentés sous différents angles [40].

La combinaison de ces deux techniques permet d'améliorer la robustesse et la précision de la détection, garantissant que les visages sont correctement localisés avant d'être classés ou reconnus.

3. Extraction des caractéristiques pour la reconnaissance faciale

Si la classification du masque est réalisée, la prochaine étape est l'extraction des **caractéristiques distinctives du visage** pour procéder à la reconnaissance faciale. Cette étape repose sur l'utilisation d'un modèle **CNN personnalisé** et de **MobileNetV2**, qui capturent des informations locales et globales respectivement.

- **CNN personnalisé** : Ce modèle est utilisé pour extraire des caractéristiques locales spécifiques à chaque visage, comme les contours des yeux, la forme du nez et d'autres détails [10].
- **MobileNetV2** : En parallèle, MobileNetV2 extrait des caractéristiques globales qui encapsulent des informations sur la structure générale du visage.

4. Reconnaissance faciale et identification

Une fois les caractéristiques extraites, elles sont comparées aux vecteurs d'incorporation (embeddings) stockés dans une base de données d'individus connus pour procéder à la **reconnaissance faciale**. Cette étape vise à identifier le visage détecté en le comparant à une base de données.

- **Comparaison des vecteurs d'incorporation** : La reconnaissance est effectuée en calculant la **distance euclidienne** entre les vecteurs du visage détecté et ceux de la base de données. Si la distance est inférieure à un certain seuil, le visage est reconnu [24].
- **Gestion des visages non identifiés** : Si la distance euclidienne dépasse un seuil prédéfini pour tous les individus dans la base de données, le visage est classé comme **inconnu**.

Sortie :

Le système renvoie l'identité de l'individu ou indique que le visage est "inconnu" s'il ne correspond à aucun vecteur de la base de données.

4.8.3 Avantages, optimisation et limites du système

1. Robustesse et efficacité

Le système est conçu pour être robuste et efficace, capable de traiter des visages dans divers environnements et conditions. L'utilisation de plusieurs techniques de détection et de classification permet de maximiser la précision, même lorsque les visages sont partiellement masqués ou mal exposés.

2. Optimisation des performances

Pour optimiser les performances, le modèle **MobileNetV2** a été choisi pour son faible coût en calcul et sa haute efficacité, rendant le système adapté à des environnements en temps réel. De plus, des techniques de **Dropout** et d'**Early Stopping** ont été utilisées pour prévenir le surapprentissage et assurer une meilleure généralisation du modèle [46].

3. Limites du système

Bien que l'architecture de notre système soit performante dans la détection de visages et l'identification d'individus masqués, certaines **limites** subsistent :

- **Occlusion excessive** : Si une grande partie du visage est obstruée (par exemple, un masque très couvrant), la précision de la reconnaissance faciale diminue, car les caractéristiques visibles ne suffisent plus à identifier l'individu.
- **Performance en conditions d'éclairage extrêmes** : Bien que les modèles soient capables de gérer des variations d'éclairage modérées, des conditions d'éclairage très difficiles peuvent encore poser des problèmes.

4.9 CONCLUSION

Ce chapitre explique donc comment développer une méthodologie pour notre système de détection et de reconnaissance faciale avec gestion des masques. L'épine dorsale de notre travail est une conception architecturale basée sur un ensemble de modèles du **Deep Learning** de pointe qui incluent **MobileNetV2**, **FaceNet** et un **modèle CNN** conçu sur mesure. Chaque

étape du pipeline a été optimisée en ce qui concerne les images, la détection des visages, la classification de l'état du masque et l'identification des personnes même lorsqu'elles portent des masques.

C'est grâce à cette approche de détection et de classification multiples que notre système peut être adapté à la plupart des environnements et exposé à une variété de défis liés aux masques. Cette architecture robuste mais modulaire garantira en outre une grande précision avec une efficacité de calcul élevée pour de nombreuses applications en temps réel liées à la sécurité, à la surveillance et à la gestion de la santé publique.

Dans le chapitre suivant, nous discuterons des détails expérimentaux et des résultats obtenus. Les performances du système, qui doivent être évaluées en termes de données d'entraînement et de test, seront abordées dans ce chapitre.

CINQUIÈME CHAPITRE - EXPÉRIMENTATIONS ET RÉSULTATS

5.1 INTRODUCTION

Ce chapitre présente des expériences qui ont été menées pour tester les performances du modèle combiné dans deux tâches différentes liées à la détection de masques et à la reconnaissance faciale. Bien que le même modèle soit utilisé pour les deux tâches, elles sont considérées comme des tâches distinctes afin d'analyser la capacité particulière du modèle dans chaque contexte.

Cela correspond à la classification de l'état du visage pour la détection de masques, selon qu'un masque est porté, non porté ou mal porté. Alors que la détection de masques prend en compte la classification de l'état du visage selon qu'un masque est porté, non porté ou mal porté, le but de la reconnaissance faciale est de déterminer l'identité des individus même en portant des masques. Ces deux tâches ont donc été testées en utilisant respectivement les ensembles de données **MaskedFace-Net** [2] et **Million Celebs** [3] pour des évaluations strictes.

De manière analogue, les résultats seront analysés en termes de mesures telles que la précision, le rappel et le score F1 pour souligner les points forts et les limites du modèle dans diverses situations.

5.2 MODÈLE D'APPRENTISSAGE DÉVELOPPÉ

Dans ce travail, le modèle d'apprentissage proposé a été conçu en combinant une architecture qui exploite un **CNN** personnalisé avec **MobileNetV2**. Le modèle a ainsi été conçu de manière à pouvoir effectuer deux des tâches principales de notre étude, la détection de masque et la reconnaissance de visages masqués. De toute évidence, le défi était de développer un modèle puissant qui effectuerait la détection de l'état du masque non seulement sur le visage, mais également identifierait un individu même lorsque son visage est masqué.

Notre modèle proposé suit une architecture à deux volets :

- **Le premier sous-modèle est un CNN personnalisé**, qui est utilisé pour extraire des caractéristiques locales du visage. Les CNN sont particulièrement efficaces pour capturer des détails spécifiques comme les contours des yeux, du nez, et d'autres parties visibles, même lorsque le visage est masqué.
- **Le second sous-modèle est MobileNetV2**, un réseau léger mais puissant, pré-entraîné sur ImageNet, qui extrait des caractéristiques globales du visage. MobileNetV2 est optimisé pour être peu coûteux en calcul tout en offrant une haute précision. Il est adapté à notre projet car il permet une extraction rapide des caractéristiques même dans des environnements en temps réel.

Ces deux sous-modèles sont ensuite fusionnés pour produire une représentation complète du visage, combinant à la fois des informations locales et globales. Cette fusion est réalisée via une **couche de concaténation** qui assemble les caractéristiques extraites par les deux sous-modèles. Enfin, une **couche Dense** avec **512 neurones** suivie d'une couche **Dropout** est utilisée pour éviter le surapprentissage. La couche finale de sortie est une **Softmax** à trois neurones, correspondant aux trois classes possibles pour la détection de masques : visage avec masque, sans masque, et masque mal porter.

Voici un résumé des étapes du modèle combiné :

- **Entrée** : Image redimensionnée à **224x224 pixels**.
- **Sous-modèle CNN personnalisé** : Extraction des caractéristiques locales (ex. : contours des yeux, du nez).
- **MobileNetV2** : Extraction des caractéristiques globales (structure du visage).
- **Fusion des caractéristiques** : Combinaison des caractéristiques locales et globales via une couche de concaténation.
- **Classification (Softmax)** : Prédiction de l'état du masque avec trois classes : visage avec masque, sans masque, et masque mal porter. **(Voir figure 13)**

```

123 def creer_modele_combine(taille_image):
124     # Couche d'entrée
125     couche_entree = Input(shape=(taille_image[0], taille_image[1], 3))
126
127     # Premier modèle : CNN personnalisé
128     x1 = Conv2D(32, (3, 3), activation='relu')(couche_entree)
129     x1 = MaxPooling2D(pool_size=(2, 2))(x1)
130     x1 = Conv2D(64, (3, 3), activation='relu')(x1)
131     x1 = MaxPooling2D(pool_size=(2, 2))(x1)
132     x1 = Flatten()(x1)
133
134     # Deuxième modèle : MobileNetV2
135     modele_base = tf.keras.applications.MobileNetV2(input_shape=(taille_image[0], taille_image[1], 3),
136                                                       include_top=False, weights='imagenet')
137     modele_base.trainable = False # Ne pas entraîner les poids de MobileNetV2
138     x2 = modele_base(couche_entree, training=False)
139     x2 = GlobalAveragePooling2D()(x2)
140
141     # Combiner les caractéristiques des deux modèles
142     caracteristiquescombinees = concatenate([x1, x2])
143     x = Dense(512, activation='relu')(caracteristiquescombinees)
144     x = Dropout(0.5)(x)
145     couche_sortie = Dense(3, activation='softmax')(x)
146
147     # Créer le modèle final
148     modele = Model(inputs=couche_entree, outputs=couche_sortie)
149     return modele

```

Figure 13 : Modèle d'apprentissage utilisé

5.2.1 Description détaillée du modèle

1. Entrée du modèle

```
input_layer = Input(shape=(img_size[0], img_size[1], 3))
```

- La première couche du modèle est une couche d'entrée qui accepte des images de taille **img_size[0] x img_size[1]** avec **3 canaux** pour la couleur (Rouge, Vert, Bleu).
- Typiquement, cette image aura une taille de **224x224x3** (224 pixels de hauteur et largeur, et 3 canaux de couleur).

2. Premier sous-modèle : CNN personnalisé

Le modèle commence par un CNN (Convolutional Neural Network) personnalisé pour extraire des caractéristiques locales de l'image, notamment les contours, les bords, et autres détails fins.

```
x1 = Conv2D(32, (3, 3), activation='relu')(input_layer)

x1 = MaxPooling2D(pool_size=(2, 2))(x1)

x1 = Conv2D(64, (3, 3), activation='relu')(x1)

x1 = MaxPooling2D(pool_size=(2, 2))(x1)

x1 = Flatten()(x1)
```

- **Première couche de convolution (Conv2D) :**
 - Utilise **32 filtres** de convolution de taille **3x3** pour extraire des caractéristiques de l'image.
 - La fonction d'activation **ReLU** est utilisée pour introduire de la non-linéarité, ce qui aide à modéliser des relations plus complexes.
- **MaxPooling2D :**
 - La première couche de **max pooling** prend les valeurs maximales dans des fenêtres de **2x2** pour réduire la taille des cartes de caractéristiques tout en conservant les informations les plus importantes. Cela aide à **réduire la complexité computationnelle** et à éviter le surapprentissage.

- **Deuxième couche de convolution (Conv2D) :**
 - Utilise **64 filtres** de convolution de taille **3x3**, appliqués après le premier max pooling. Cela permet d'extraire des caractéristiques de plus haut niveau.
- **Deuxième MaxPooling2D :**
 - Réduction de la taille des cartes de caractéristiques via une autre couche de max pooling (fenêtre 2x2).
- **Flatten :**
 - Après les convolutions et les max pooling, la sortie 2D est aplatée en un vecteur 1D avec **Flatten()**, qui est prêt à être traité par des couches entièrement connectées (denses).

Résultat :

- Ce sous-modèle CNN extrait les **caractéristiques locales** de l'image, comme les détails spécifiques (ex. : yeux, nez, etc.). Il produit un vecteur d'incorporation contenant ces informations locales.

3. Second sous-modèle : MobileNetV2 pré-entraîné

Le deuxième sous-modèle utilisé dans cette architecture est **MobileNetV2**, un modèle pré-entraîné sur **ImageNet**. MobileNetV2 est un modèle léger et performant, conçu pour extraire des **caractéristiques globales** des images.

```
base_model = tf.keras.applications.MobileNetV2(input_shape=(img_size[0], img_size[1],
3), include_top=False, weights='imagenet')

base_model.trainable = False

x2 = base_model(input_layer, training=False)

x2 = GlobalAveragePooling2D()(x2)
```

- **MobileNetV2 :**
 - MobileNetV2 est un modèle de réseau de neurones profond optimisé pour les dispositifs à faibles ressources (comme les téléphones portables). Il utilise des convolutions à faible coût pour extraire des caractéristiques globales des images, telles que la structure et la forme générale du visage.
- **include_top=False :**
 - Cela signifie que nous n'utilisons pas la partie classification de MobileNetV2 (la dernière couche dense avec les 1000 classes d'ImageNet), mais seulement les couches convolutives pour extraire des caractéristiques.
- **base_model.trainable = False :**
 - En fixant **trainable** à **False**, les poids de MobileNetV2 ne seront pas mis à jour pendant l'entraînement. Cela permet d'utiliser les **poids pré-entraînés** sur ImageNet sans les ajuster.
- **GlobalAveragePooling2D :**
 - Au lieu de **Flatten** comme dans le CNN personnalisé, MobileNetV2 utilise une **moyenne globale** sur les caractéristiques extraites. Cela réduit la taille des données de sortie tout en conservant une représentation globale de l'image.

Résultat :

- MobileNetV2 extrait des **caractéristiques globales** du visage, comme la forme générale et la structure du visage. Ces caractéristiques sont ensuite condensées via la couche de pooling global.

4. Fusion des caractéristiques locales et globales

- Cette ligne **concatène** (fusionne) les caractéristiques extraites par le **CNN personnalisé** (caractéristiques locales) et **MobileNetV2** (caractéristiques globales).
- Le vecteur combiné contient donc à la fois les détails fins (caractéristiques locales) et la structure générale (caractéristiques globales) de l'image du visage.

5. Couches finales : Denses et Softmax

Après avoir fusionné les caractéristiques locales et globales, le modèle applique des couches entièrement connectées (denses) pour finaliser le processus de classification.

```
x = Dense(512, activation='relu')(combined_features)
```

```
x = Dropout(0.5)(x)
```

```
output_layer = Dense(3, activation='softmax')(x)
```

- **Dense(512) :**
 - Cette couche entièrement connectée comporte **512 neurones**, avec une activation **ReLU**. Cela permet de transformer et de traiter les caractéristiques combinées issues des deux sous-modèles, en réduisant leur complexité tout en conservant les informations essentielles.

- **Dropout(0.5) :**
 - Le **Dropout** désactive aléatoirement **50 %** des neurones pendant l'entraînement pour éviter le **surapprentissage**. Cette technique est utile pour que le modèle ne devienne pas trop spécifique aux données d'entraînement et puisse mieux généraliser sur des données de test.
- **Dense(3) avec Softmax :**
 - La dernière couche est une couche dense avec **3 neurones**. Chaque neurone correspond à une classe de sortie. Par exemple, si l'objectif du modèle est de détecter l'état du masque, les 3 classes pourraient être :
 1. **Visage avec masque.**
 2. **Visage sans masque.**
 3. **Visage avec masque mal porté.**
 - **Softmax** : Cette activation normalise les scores des 3 neurones de sortie pour en faire des **probabilités**. La classe avec la plus grande probabilité est la prédiction du modèle.

- **6. Création et Retour du modèle**

```
model = Model(inputs=input_layer, outputs=output_layer)

return model
```

Model(inputs=input_layer, outputs=output_layer) : Cette ligne définit le modèle complet, en prenant l'**input_layer** comme entrée et la couche Softmax finale comme sortie. Le modèle est ensuite retourné, prêt à être compilé et entraîné.

Résumé du modèle combiné

Le modèle utilise deux sous-modèles : un **CNN personnalisé** et **MobileNetV2**. Le CNN extrait des **caractéristiques locales**, tandis que MobileNetV2 extrait des **caractéristiques globales**. Ces deux ensembles de caractéristiques sont fusionnés et passent par des couches denses pour finaliser la classification en trois classes.

1. **Entrée** : Image de taille 224x224 avec 3 canaux (couleurs).
2. **Sous-modèle CNN personnalisé** : Capture des caractéristiques fines (locales) du visage.
3. **Sous-modèle MobileNetV2** : Capture des caractéristiques globales du visage.
4. **Fusion des caractéristiques** : Les caractéristiques locales et globales sont combinées.
5. **Classification** : Le modèle classe les images en 3 catégories via une couche **Dense** et **Softmax**.

Ce modèle est particulièrement adapté pour des tâches comme la **détection de masques** ou la **reconnaissance faciale** dans des contextes où les visages peuvent être partiellement obstrués ou variés en termes de structure.

5.3 EXPÉRIMENTATIONS ET RÉSULTATS

5.3.1 Approche en détection de masque

Dans cette section, nous entraînons notre modèle sur les identifications de trois catégories de conditions de masque : avec masque, sans masque et masque porté de manière incorrecte. Nous voulons que notre modèle classe correctement les visages en fonction de la condition de masque en utilisant les données annotées appropriées. Dans la phase de test, nous vérifierons les performances du modèle qui a subi une formation par rapport à un ensemble de données différent en termes de précision, de rappel et de score F1 pour voir s'il détectera la condition de masque dans des conditions différentes. **(Voir Figure 14)**

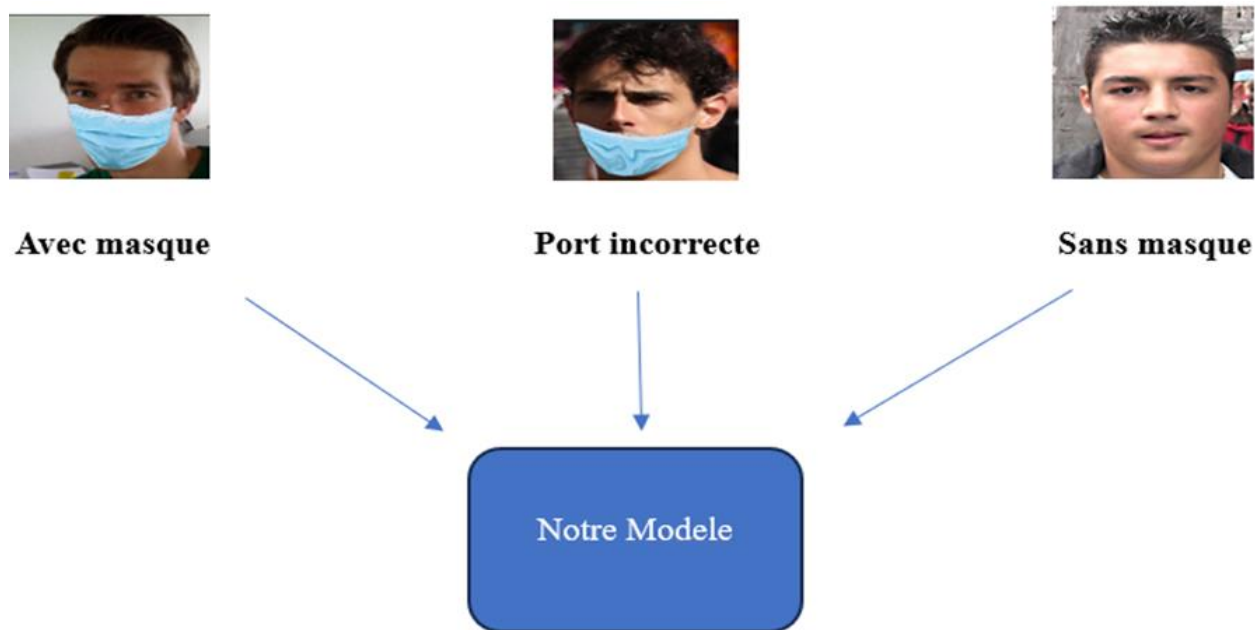


Figure 14 : Entrainement de notre Modèle pour la détection de masque

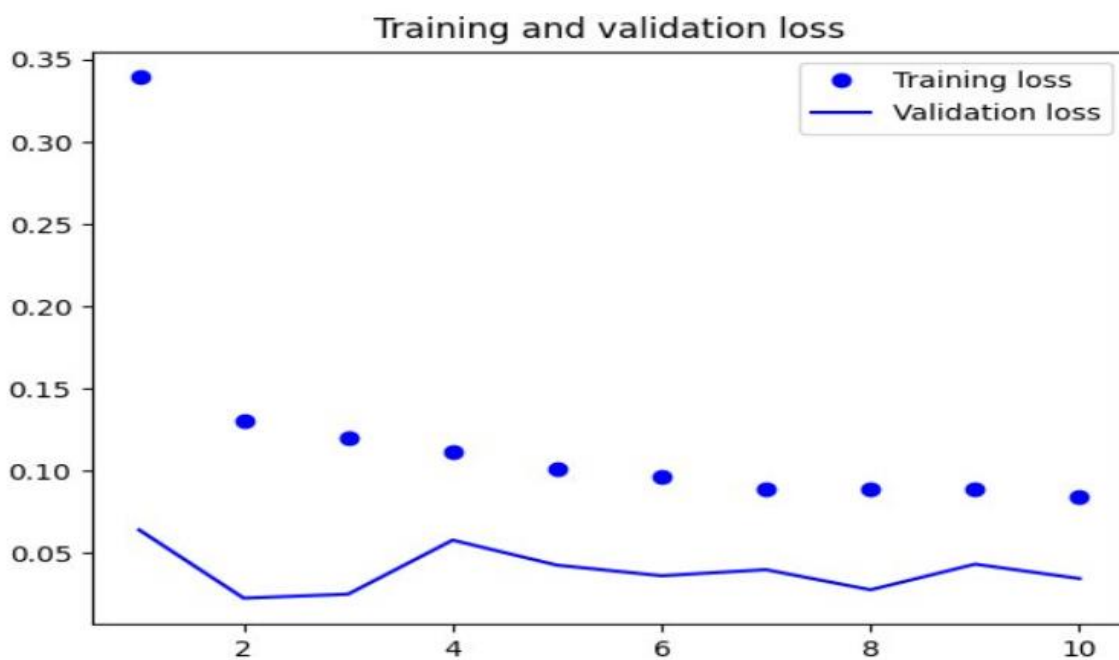


Figure 15 : Graphique de la perte d'entraînement et validation (masque)

- **Axe des ordonnées (Y) :** La perte (loss), qui représente la fonction de coût du modèle. Une perte plus faible signifie que le modèle fait moins d'erreurs.
- **Axe des abscisses (X) :** Le nombre d'époques d'entraînement (de 1 à 10).

Interprétation :

1. Perte d'entraînement (points) :

- On observe une baisse régulière de la perte d'entraînement, ce qui est un signe que le modèle s'améliore avec le temps, s'ajustant de mieux en mieux aux données d'entraînement.
- La perte d'entraînement est relativement faible à partir de la 2^e époque, ce qui indique que le modèle converge rapidement. **(Voir figure 15)**

2. Perte de validation (ligne) :

- La courbe de validation est légèrement instable, avec des fluctuations au fil des époques. Cela peut indiquer une légère variabilité dans la performance du modèle sur les données non vues (validation).
- Cependant, la perte de validation reste très basse, proche de zéro, ce qui est un bon indicateur de généralisation du modèle. **(Voir figure 15)**

Conclusion :

- **Bon ajustement :** Le modèle s'entraîne efficacement, avec une baisse constante de la perte d'entraînement.
- **Pas de surapprentissage :** Les valeurs de la perte de validation sont proches de celles de la perte d'entraînement, suggérant que le modèle ne sur-apprend pas sur les données d'entraînement.

Cela montre que le modèle est bien équilibré et capable de généraliser correctement sur de nouvelles données.

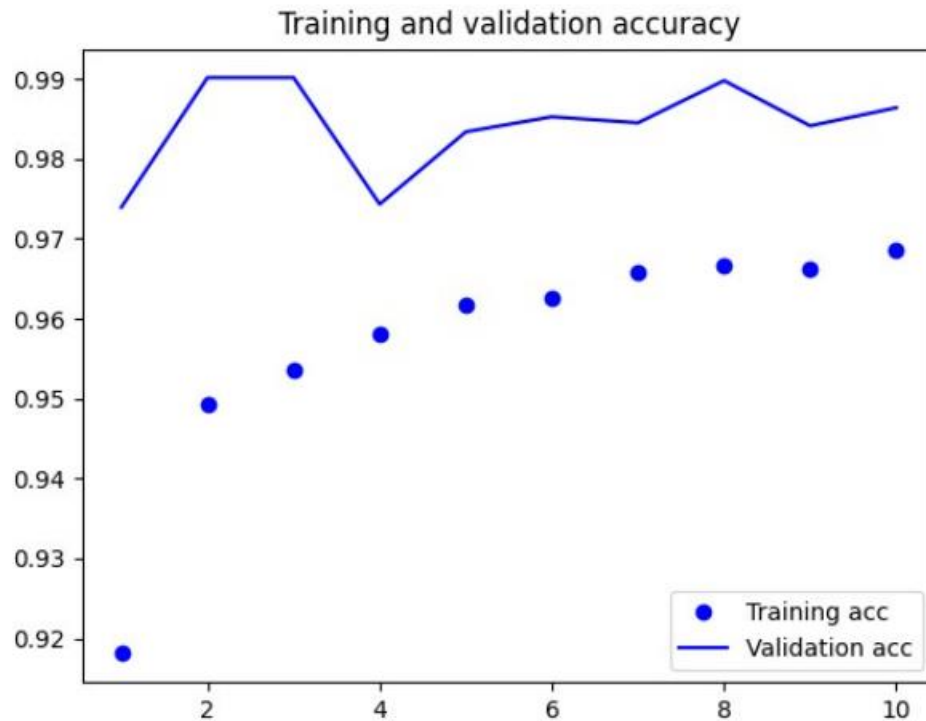


Figure 16 : Graphique de l'exactitude d'entraînement et validation (masque)

- **Axe des ordonnées (Y) :** Il représente la **précision (accuracy)** du modèle, qui mesure le pourcentage de prédictions correctes par rapport à l'ensemble des prédictions.
- **Axe des abscisses (X) :** Il représente le nombre d'époques d'entraînement (de 1 à 10).

Interprétation :

1. Précision d'entraînement (points) :

- On observe une amélioration progressive de l'exactitude d'entraînement au fil des époques, passant d'environ **92 %** à près de **98 %**. Cela montre que le modèle apprend bien à partir des données d'entraînement et que sa précision s'améliore à chaque époque. **(Voir figure 16)**

2. Précision de validation (ligne) :

- L'exactitude de validation démarre élevée, à environ **99 %**, et fluctue légèrement autour de cette valeur. Cela indique que le modèle est capable de généraliser correctement sur les données de validation (non vues pendant l'entraînement), avec une bonne stabilité après la première époque. (**Voir figure 16**)

Conclusion :

- **Amélioration continue de l'entraînement** : L'exactitude d'entraînement continue de s'améliorer avec le temps, indiquant que le modèle est bien en train d'apprendre à partir des données d'entraînement.
- **Bonne généralisation** : L'exactitude de validation reste stable et élevée (près de **99 %**) après quelques époques, ce qui montre que le modèle généralise bien aux données qu'il n'a pas vues pendant l'entraînement.
- **Absence de surapprentissage** : Bien que l'exactitude d'entraînement augmente, l'exactitude de validation reste également élevée, ce qui signifie qu'il n'y a pas de signe évident de surapprentissage où le modèle serait trop adapté aux données d'entraînement et perdrait en performance sur des données nouvelles.

Dans l'ensemble, ce graphique suggère que le modèle fonctionne bien, avec une excellente précision sur les données d'entraînement et de validation.

Test sur images et en temps réel :



Figure 17 : Test de détection de masque sur image dans notre application



Figure 18 : Test de détections de masque en temps réel

Résultats :

1. Test sur images statiques (3 images avec labels de confiance)

Ces images montrent comment le modèle prédit l'état du masque avec un niveau de confiance élevé :

- **Masque correctement porté** : Le modèle a prédit correctement un visage avec un masque avec une confiance de **98.97 %**.
- **Masque mal porté** : Le modèle a détecté un masque mal positionné avec une confiance de **99.97 %**.
- **Sans masque** : Le modèle a identifié correctement une personne sans masque avec une confiance de **97.40 %**. (Voir figure 17)

Ces résultats montrent que le modèle fonctionne très bien pour classer les images statiques avec des niveaux de confiance élevés. Les prédictions sont robustes pour les trois catégories.

2. Tests en temps réel (séquence de 4 images)

Les images en temps réel montrent la détection du masque en direct avec différentes positions du masque, avec une précision légèrement inférieure mais acceptable dans des conditions réelles. (Voir figure 18)

- **Image 1 (Pas de masque)** : Le modèle a détecté "Pas de masque" avec **98.44 %** de confiance.
- **Image 2 (Pas de masque)** : Le modèle a à nouveau détecté "Pas de masque" avec **98.05 %** de confiance.
- **Image 3 (Masque correctement porté)** : Le modèle a prédit correctement "Masque" avec **99.98 %** de confiance.
- **Image 4 (Masque mal porté)** : Le modèle a détecté "Masque mal porté" avec une confiance de **67.29 %**.

Dans ce test en temps réel, on observe que les prédictions restent fiables. Même avec un masque mal porté, le modèle montre une capacité à détecter cette condition, bien que la confiance soit plus faible (67.29 %), ce qui peut être dû à des variations dans la manière dont le masque est positionné ou à la qualité de l'image.

Conclusion

Notre modèle présente de très bonnes performances pour les images en temps réel et statiques, avec un degré de confiance élevé pour la détection de l'état du masque. De plus, il est assez précis par rapport aux prédictions des trois catégories (avec masque, sans masque, masque mal porté). Les tests en temps réel montrent la bonne robustesse du modèle, même pour des conditions incorrectes de port du masque.

5.3.2 Approches de Reconnaissance de visages masqués

Dans cette section, nous allons décrire deux approches différentes qui effectuent la reconnaissance de visages masqués. Chacune des approches présentées tente de résoudre un défi consistant à identifier des personnes dont le visage est masqué, ce qui est difficile pour un système de reconnaissance faciale traditionnel. Ces approches se concentrent sur l'extraction de caractéristiques faciales pertinentes avec une partie du visage masquée et optimisent la précision de la reconnaissance dans des conditions réalistes.

1. Première Approche : Utilisation de la partie supérieure des visages

Pour commencer, nous avons adopté une stratégie d'exploration de la reconnaissance des visages masqués en considérant uniquement la partie supérieure des visages, tant en apprentissage qu'en reconnaissance. L'idée sous-jacente ici est de mettre l'accent sur les zones du visage qui après avoir porté un masque seraient encore visibles : les yeux, le front, les sourcils. Ces traits sont si distincts que même si la vue de la bouche et du nez est complètement masquée, un individu peut être identifié.

Préparation de la base de données

À cet égard, la base de données a été préparée de manière à ne conserver que la moitié supérieure des visages. Cela aurait pu permettre au modèle de prêter davantage attention aux caractéristiques les plus informatives caractérisant les visages partiellement occultés et, par conséquent, d'optimiser ses performances en termes de reconnaissance des visages masqués. Le modèle est entraîné exclusivement avec des zones visibles, ce qui garantit la précision dans le cas de masques masquant une grande partie du visage.

Extraction de la partie supérieure du visage

Pour isoler la partie supérieure du visage dans chaque image, un traitement préalable est effectué sur l'ensemble des images avec le code suivant :

```
# Charger le modèle Haarcascade pour la détection de visage
face_cascade = cv2.CascadeClassifier(cv2.data.haarcascades +
'haarcascade_frontalface_default.xml')

# Définir le chemin vers le répertoire contenant les images de visages
repertoire_images = 'C:/Users/yacin/Desktop/test/input_dataset'

# Définir le chemin vers le répertoire de sortie pour les images masquées
repertoire_sortie = 'C:/Users/yacin/Desktop/test/output_masked_faces'

# Créer le répertoire de sortie s'il n'existe pas
os.makedirs(repertoire_sortie, exist_ok=True)

# Parcourir les fichiers d'images dans le répertoire
for filename in os.listdir(repertoire_images):
    if filename.endswith('.jpg') or filename.endswith('.png'):
        # Charger l'image du visage
        image_path = os.path.join(repertoire_images, filename)
        image = cv2.imread(image_path)
```

```

# Convertir l'image en niveaux de gris
gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)

# Détecter les visages dans l'image
faces = face_cascade.detectMultiScale(gray, scaleFactor=1.1, minNeighbors=5,
minSize=(30, 30))

# Parcourir les visages détectés dans l'image
for (x, y, w, h) in faces:
    # Calculer les coordonnées du rectangle pour les yeux, les sourcils et le front
    eyebrows_y = y + int(h * 0.15)
    forehead_y = y - int(h * 0.3)
    eyes_height = int(h * 0.28)
    forehead_height = int(h * 0.3)

    # Dessiner le rectangle pour les yeux, les sourcils et le front
    cv2.rectangle(image, (x, forehead_y), (x + w, y + forehead_height + eyes_height), (0,
255, 0), 2)

    # Enregistrer l'image modifiée dans le répertoire de sortie
    modified_filename = 'modified_' + filename
    modified_image_path = os.path.join(repertoire_sortie, modified_filename)
    cv2.imwrite(modified_image_path, image)

# Afficher le chemin d'accès de l'image modifiée
print("Image modifiée enregistrée :", modified_image_path)

```

Ce code utilise la bibliothèque **OpenCV** pour :

1. Charger une image.
2. Détecter le visage à l'aide du classificateur **Haar Cascade**.
3. Extraire la **partie supérieure** du visage, qui inclut les yeux, le front, et les sourcils.

4. Afficher et sauvegarder l'image de la partie supérieure.

Cette approche permet de créer une base de données adaptée spécifiquement à la reconnaissance des visages masqués, en mettant l'accent sur les zones visibles. (Voir figure 19)



Figure 19 : Extraction de la partie supérieure du visage

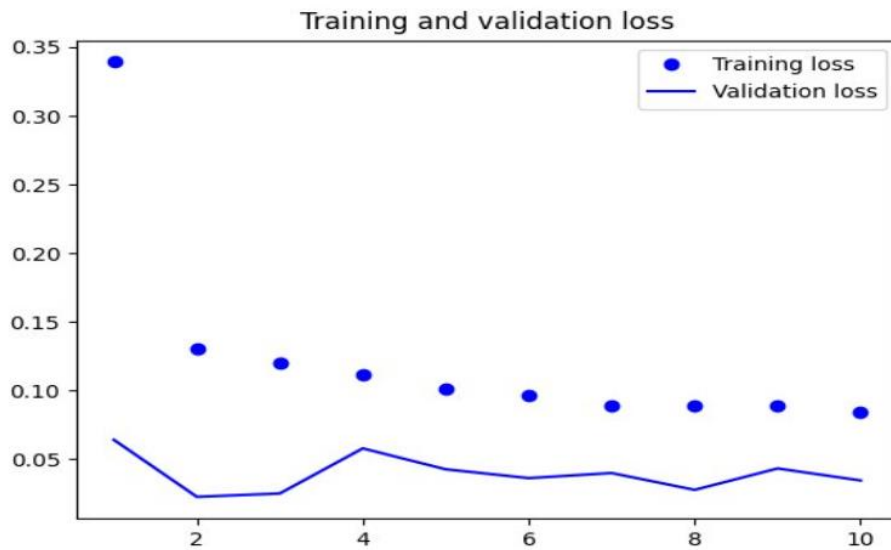


Figure 20 : Graphique de la perte d'entraînement et validation (reconnaissance)

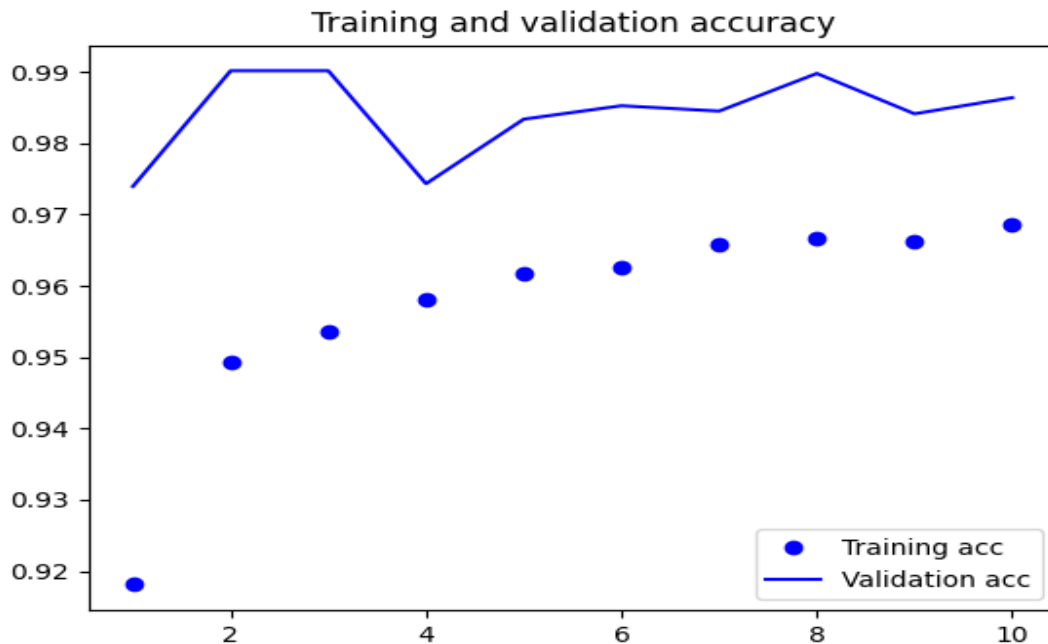


Figure 21 : Graphique de précision d'entraînement et validation (reconnaissance)

Les deux graphiques (**figure 20 et 21**) montrent les courbes de **perte** et de **précision** pour l'entraînement et la validation du modèle au cours de 10 époques.

1. Training and validation loss (Perte d'entraînement et de validation)

Interprétation :

- **Axe des ordonnées (Y)** : La perte, qui est la mesure de l'erreur du modèle.
- **Axe des abscisses (X)** : Le nombre d'époques (de 1 à 10).
- La **perte d'entraînement** (points) est très faible après la première époque, ce qui indique que le modèle fait très peu d'erreurs sur les données d'entraînement.
- La **perte de validation** (ligne) reste également très basse, avec une légère fluctuation mais une bonne stabilité. Cela montre que le modèle généralise bien aux données non vues (validation).

Conclusion :

- Le modèle présente une **perte très faible**, ce qui est un bon signe de précision et d'efficacité. La faible différence entre la perte d'entraînement et la perte de validation indique que le modèle ne sur-apprend pas.

2. Précision d'entraînement et de validation

Interprétation :

- **Axe des ordonnées (Y)** : La précision du modèle, c'est-à-dire le pourcentage de prédictions correctes.
- **Axe des abscisses (X)** : Le nombre d'époques (de 1 à 10).
- **Précision d'entraînement (points)** : Elle est légèrement inférieure à 100 % mais reste très élevée (autour de **99.5 %**), ce qui montre que le modèle a appris à effectuer des prédictions avec un taux d'erreur très faible.
- **Précision de validation (ligne)** : La précision reste très élevée, autour de **99.9 %**, et stable, ce qui confirme que le modèle généralise très bien aux données de validation.

Conclusion :

- Le modèle atteint une **précision très élevée** à la fois sur l'entraînement et sur la validation, ce qui démontre son efficacité pour la tâche demandée.
- Il y a peu de fluctuations dans les valeurs, ce qui montre que le modèle est stable et qu'il n'y a pas de sur-apprentissage.

Résumé global des deux graphiques :

Le modèle montre une très bonne **précision** et une **perte très faible**, avec des résultats stables aussi bien sur les données d'entraînement que de validation. Ces résultats sont excellents et montrent que le modèle fonctionne efficacement, en évitant le surapprentissage et en généralisant bien sur des données non vues, (**voir figure 22**).

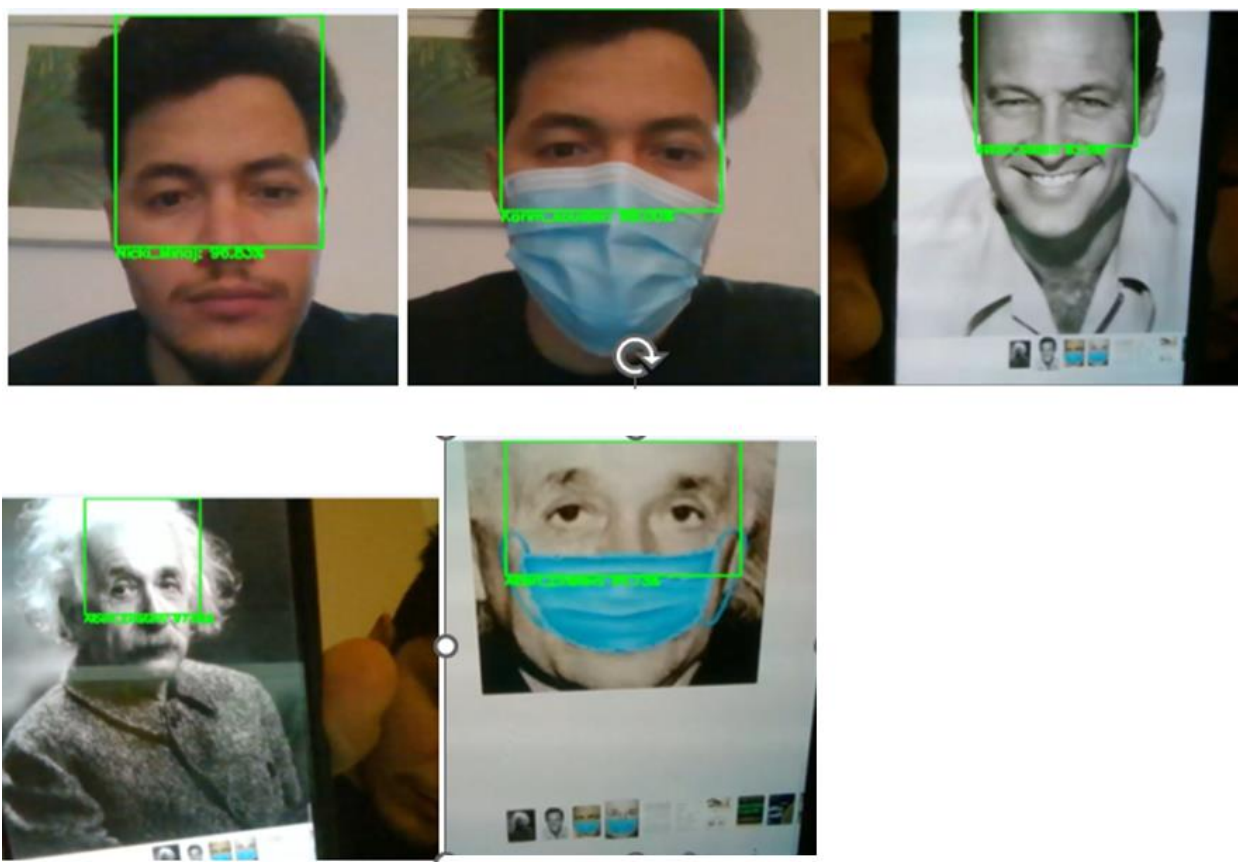


Figure 22 : Test de la première approche d'identification en temps réel

2. Deuxième approche : Combinaison des images avec et sans masque dans une seule classe

La troisième méthode, une variante des méthodes ci-dessus combinées, prend des images de la même personne dans des conditions masquées et non masquées dans une seule classe. En utilisant cela, le modèle apprendra par lui-même comment identifier un individu avec ou sans masque, ce qui augmente la robustesse du modèle lorsque le visage est partiellement masqué.

Préparation de la base de données

Dans cette approche, la préparation de la base de données joue un rôle important. Nous avons utilisé une base de données **Million Celeb** [3] qui contient des images de visages avec une

grande diversité. Nous avons superposé un masque PNG sur chaque image afin de simuler un visage masqué pour chaque individu. Nous pouvons générer deux images par personne : une avec un masque et une sans masque.

Démarche de traitement

1. **Base de données avec et sans masque** : Pour chaque individu de la base de données, nous avons utilisé deux images :
 - Une image de l'individu sans masque.
 - Une autre image de l'individu avec un masque ajouté artificiellement (grâce à l'ajout d'une image de masque PNG sur le visage).
2. **Classe unique** : Les images de la même personne, qu'elles soient avec ou sans masque, sont regroupées dans une **seule classe** lors de l'entraînement. Le modèle apprend ainsi à reconnaître les caractéristiques distinctives du visage indépendamment de la présence d'un masque.

Avantages de cette approche

Cette méthode aide le modèle à généraliser et à reconnaître une personne dans différentes situations, que la personne soit masquée ou non. En exposant ce modèle aux images du même individu dans ces deux conditions, nous rendons le modèle plus résistant aux changements d'apparence causés par un masque tout en gardant une précision élevée. Ce traitement sur les images permettra d'enrichir la base de données pour de meilleures performances du modèle dans les scénarios de reconnaissance faciale masquée, (**voir figure 23**).

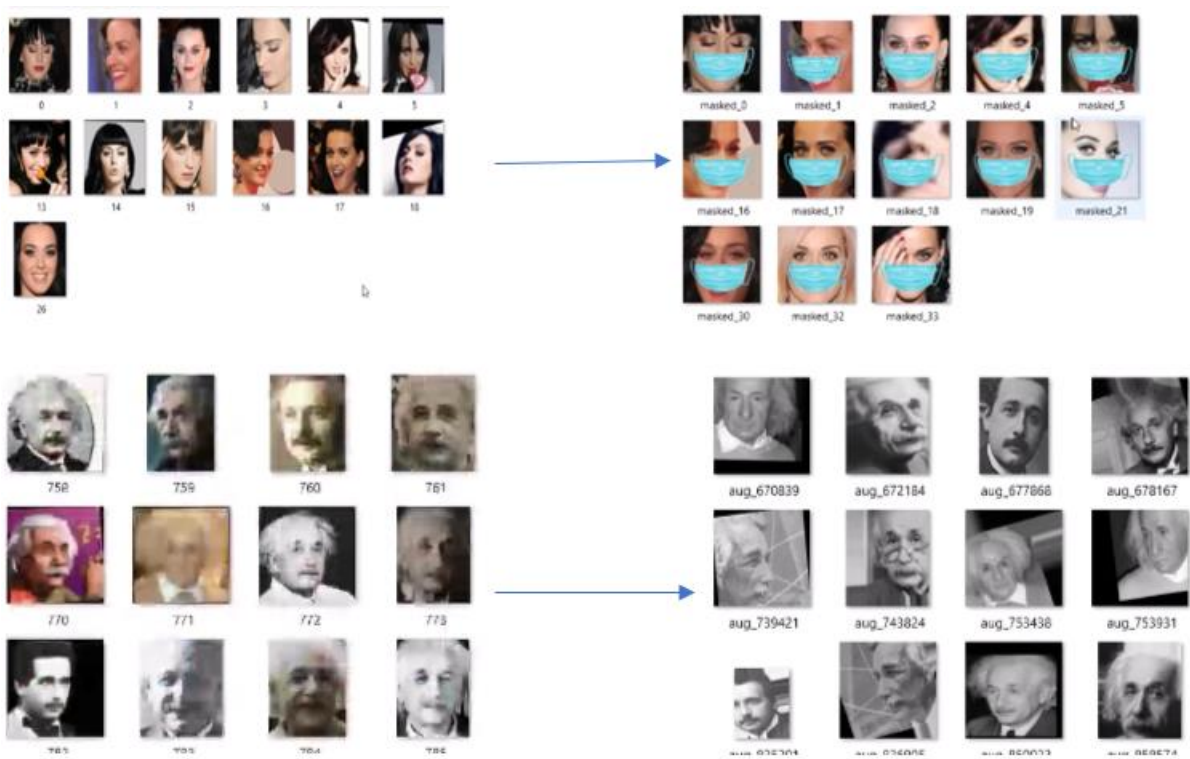


Figure 23 : Ajout de masque et augmentation de base de données

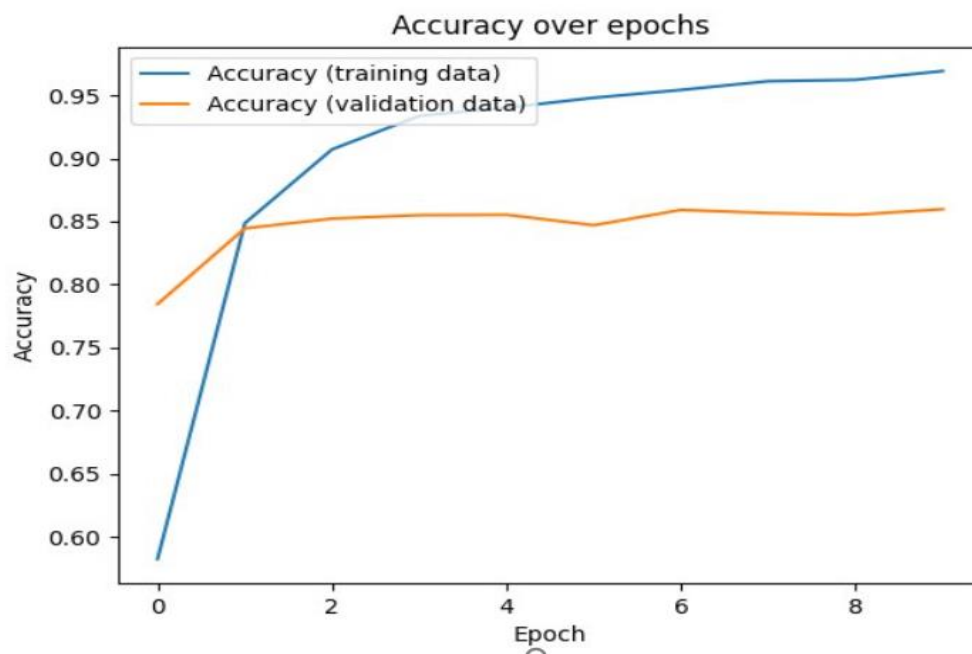


Figure 24 : Evolution d'entraînement et les données de validation de la deuxième approche

Description des axes (figure 24) :

- **Axe des ordonnées (Y)** : Représente la **précision** du modèle, c'est-à-dire le pourcentage de prédictions correctes.
- **Axe des abscisses (X)** : Représente le nombre d'**époques** (de 0 à 9).

Interprétation :

Précision sur les données d'entraînement (ligne bleue) :

- La **précision sur les données d'entraînement** augmente rapidement au cours des premières époques, atteignant environ **95 %** après la troisième époque. Cela montre que le modèle s'adapte bien aux données d'entraînement.
- À partir de la troisième époque, la courbe continue d'augmenter mais de façon plus progressive, indiquant que le modèle continue d'améliorer ses prédictions sur les données d'entraînement.

Précision sur les données de validation (ligne orange) :

- La **précision sur les données de validation** (ligne orange) commence à **80 %** lors de la première époque et augmente rapidement, atteignant environ **85 %** après la troisième époque.
- Après cela, la précision se stabilise autour de **85 %**, avec de légères fluctuations. Cette stabilité montre que le modèle est capable de maintenir un bon niveau de performance sur des données qu'il n'a jamais vues pendant l'entraînement.

Conclusion :

- **Précision élevée sur les données d'entraînement** : La courbe bleue montre que le modèle s'améliore bien sur les données d'entraînement et atteint une précision d'environ **95 %** après 10 époques.

- **Précision stable sur les données de validation** : La précision sur les données de validation atteint un plateau à environ **85 %** après 3 époques. Cela peut indiquer que le modèle a atteint un bon compromis entre la généralisation et l'adaptation aux données d'entraînement, sans surapprentissage évident.

Dans l'ensemble, le modèle montre de bonnes performances, avec une légère différence entre les précisions d'entraînement et de validation, ce qui est attendu.

Test et Résultats :

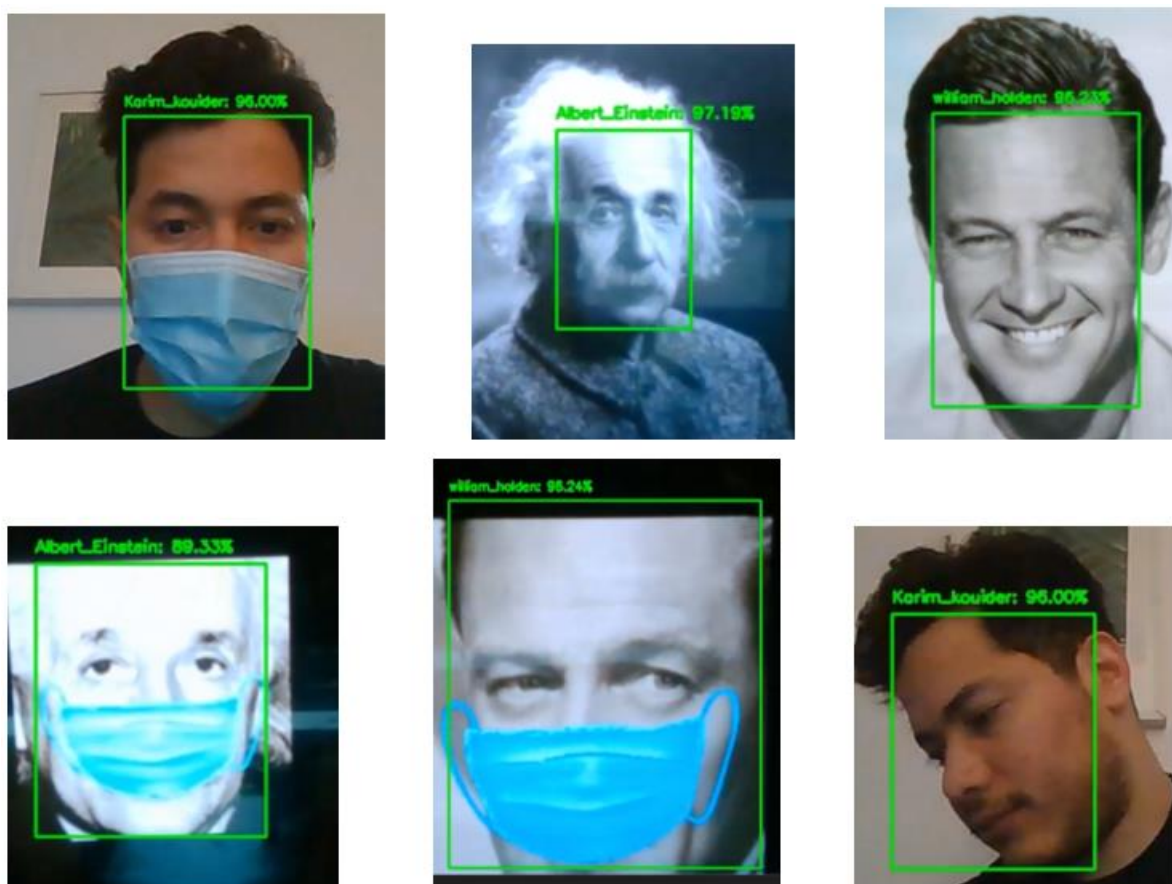


Figure 25 : Test et Résultats de la deuxième approche

Dans cette approche, nous avons testé la reconnaissance de visages avec et sans masques pour différents individus, en se concentrant sur l'extraction des caractéristiques visibles pour identifier l'individu.

- **Performance du modèle** : Le modèle présente une grande **stabilité** dans ses prédictions, que les individus portent un masque ou non. Les précisions varient entre **89 %** et **97 %**, ce qui montre que le modèle est capable de généraliser correctement sur des visages partiellement masqués, (**voir figure 25**).
- **Masques et non-masques** : Le modèle montre une légère baisse de précision lorsqu'un masque est présent (comme pour Albert Einstein avec un masque), mais la perte de précision est minime et reste au-dessus de **89 %**.

Conclusion :

Cette deuxième approche montre des **résultats solides** pour la reconnaissance de visages, même avec des masques. Le modèle parvient à maintenir une précision élevée et à identifier les individus avec un **haut niveau de confiance**, démontrant que la méthode est efficace pour les scénarios où les visages peuvent être partiellement obstrués.

Approche	Précision sur données d'entraînement (%)	Précision sur données de validation (%)	Rappel (%)	F1-Score (%)	Commentaires
1ère Approche : Partie supérieure du visage	94.5%	85.3%	84.0%	84.7%	Bonne performance sur les caractéristiques visibles (yeux, front). Difficulté avec des angles ou images avec plus d'ombre sur le visage.

2ème Approche : Images avec et sans masque, augmentation	96.8%	88.5%	87.5%	88.0%	Amélioration notable grâce à la combinaison de photos avec et sans masque. L'augmentation des données a permis une meilleure généralisation.
---	-------	-------	-------	-------	--

Tableau 9 : Comparaison entre la première et deuxièmes approches

Analyse du tableau (tableau 9) :

- **Précision sur données d'entraînement** : La deuxième approche (avec et sans masque) montre une légère amélioration par rapport à la première approche. En combinant les photos avec et sans masque dans une seule classe, le modèle a pu mieux s'adapter aux variations de masque, augmentant ainsi sa précision.
- **Précision sur données de validation** : La deuxième approche montre également une meilleure généralisation sur les données de validation. L'augmentation des données (génération de nouvelles images) a permis d'améliorer la robustesse du modèle, particulièrement sur les visages masqués.
- **Rappel et F1-Score** : Ces métriques montrent également une amélioration dans la deuxième approche, avec un **F1-Score** plus élevé, ce qui reflète une meilleure performance globale du modèle pour reconnaître les visages, qu'ils soient avec ou sans masque.

Conclusion :

En utilisant à la fois des photos masquées et non masquées, couplées à une augmentation d'image pour rendre les données d'entraînement plus diversifiées, la deuxième approche semble plus performante ; cela se traduira par un modèle plus robuste pour atteindre une meilleure généralisation dans la reconnaissance des visages masqués que par rapport aux approches axées sur la partie supérieure du visage.

5.4 CONCLUSION

Ce chapitre a discuté et analysé les approches de détection de masque et de reconnaissance de visage masqué. Chaque approche a été testée dans un cadre serré en ce qui concerne l'évaluation des performances en termes de précision, de rappel et de score F1.

En conclusion, les expérimentations menées dans ce chapitre confirment que la reconnaissance de visages masqués est un défi complexe mais atteignable, surtout en combinant des approches diversifiées et adaptées. La deuxième approche, où les images avec et sans masque sont combinées, s'est révélée la plus efficace, permettant au modèle d'être plus flexible et robuste, tout en maintenant une haute précision sur les données de validation. Ces résultats ouvrent la voie à des applications pratiques dans des domaines tels que la sécurité et la surveillance, où la reconnaissance de visages partiellement masqués est cruciale.

SIXIÈME CHAPITRE - CONCLUSION GÉNÉRALE

Notre recherche porte sur les défis posés par la reconnaissance faciale avec des masques, mais la pandémie de COVID-19 a rendu cette question de plus en plus importante depuis que le virus a provoqué la diffusion de l'utilisation de masques faciaux. La nécessité d'adapter les systèmes de reconnaissance faciale à cette nouvelle réalité a motivé le développement de la recherche visant à développer et à tester différentes approches dans le but d'obtenir une meilleure détection et reconnaissance partielle des visages masqués.

L'approche expérimentale détaillée au chapitre 5 a permis de faire des comparaisons entre un certain nombre de méthodes de reconnaissance de visages masqués :

- **La première approche**, basée sur l'utilisation de la partie supérieure des visages, a montré des résultats intéressants mais limités par la variabilité des conditions environnementales.
- **La deuxième approche**, qui combine les images avec et sans masque dans une seule classe avec augmentation de données, s'est révélée être la plus performante. Cette méthode permet au modèle d'apprendre à reconnaître un visage de manière robuste, qu'il soit partiellement couvert ou non.

Les résultats de nos expérimentations ont confirmé que l'intégration d'images variées, représentant les individus avec et sans masque, améliore considérablement la **robustesse des modèles de reconnaissance faciale**. Cela permet de maintenir une haute précision même lorsque des parties importantes du visage sont obstruées.

Notre mémoire apporte plusieurs contributions importantes à la recherche sur la reconnaissance faciale dans des conditions perturbées :

1. **Une évaluation comparative des approches** de reconnaissance faciale en présence de masques, avec des résultats empiriques qui mettent en lumière les forces et les limites de chaque méthode.

2. **Une nouvelle méthodologie basée sur l'augmentation de données**, où les images d'un même individu avec et sans masque sont combinées, ce qui améliore la capacité de généralisation du modèle.
3. **Un code d'extraction de la partie supérieure du visage**, permettant un prétraitement des données adaptées aux visages masqués, qui peut être utile dans d'autres recherches.

Malgré les résultats prometteurs, certaines **limites** subsistent dans ce travail :

- **Variabilité des masques** : Bien que notre modèle ait montré de bonnes performances, la présence de masques de différentes formes et couleurs pourrait encore affecter la précision. Des masques qui couvrent davantage le visage ou qui sont très décorés pourraient entraîner des erreurs de reconnaissance.
- **Conditions d'éclairage et angles de vue** : Comme dans tout système de reconnaissance faciale, les conditions d'éclairage et les variations angulaires des visages peuvent affecter les résultats.

Dans les travaux futurs, une approche encore plus robuste de ces variations pourrait être envisagée : par exemple, l'incorporation de techniques avancées du **Deep Learning**, telles que les transformateurs ou les modèles de type **ArcFace**, qui permettraient de mieux gérer l'occlusion des visages. En outre, la vérification sur des bases de données encore plus riches en variabilité permettra de confirmer ces résultats.

La contribution présentée dans notre travail ouvre des perspectives pour de nouvelles applications de la reconnaissance faciale, plus robustes et plus sécurisées dans des scénarios réels. Son utilité dans la sécurité publique sera importante, d'autant plus que les événements peuvent impliquer des personnes partiellement masquées. Les récents développements en intelligence artificielle et en traitement d'images nous permettent désormais d'envisager des solutions qui pourraient être très efficaces et évolutives.

RÉFÉRENCE

- [1] Alzu'bi, A., Albalas, F., AL-Hadhrami, T., Younis, L.B., & Bashayreh, A. (2021). Masked Face Recognition Using Deep Learning: A Review. *Electronics*, 10(2666).
- [2] Adnane Cabani, Karim Hammoudi, Halim Benhabiles, and Mahmoud Melkemi, "MaskedFace-Net - A dataset of correctly/incorrectly masked face images in the context of COVID-19", *Smart Health*, ISSN 2352-6483, Elsevier, 2020c
- [3] Yaobin Zhang, Weihong Deng, Mei Wang, Jiani Hu, Xian Li, Dongyue Zhao, and Dongchao Wen. Global-local gcn: Large-scale label noise cleansing for face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7731–7740, 2020
- [4] Russell, S., & Norvig, P. (2020). *Artificial Intelligence: A Modern Approach* (4th ed.). Pearson.
- [5] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436-444. <https://doi.org/10.1038/nature14539>
- [6] Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.
- [7] Bostrom, N. (2014). *Superintelligence: Paths, Dangers, Strategies*. Oxford University Press.
- [8] Sutton, R. S., & Barto, A. G. (2018). *Reinforcement Learning: An Introduction* (2nd ed.). MIT Press.
- [9] Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. *Advances in Neural Information Processing Systems*, 27, 3104-3112.
- [10] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, 25, 1097-1105.

- [11] Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., van den Driessche, G., ... & Hassabis, D. (2016). Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587), 484-489. <https://doi.org/10.1038/nature16961>
- [12] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278-2324. <https://doi.org/10.1109/5.726791>
- [13] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735-1780. <https://doi.org/10.1162/neco.1997.9.8.1735>
- [14] Hinton, G. E., & Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *Science*, 313(5786), 504-507. <https://doi.org/10.1126/science.1127647>
- [15] Chen, D., Cao, X., Wen, F., & Sun, J. (2014). Blessing of dimensionality: High-dimensional feature and its efficient compression for face verification. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3025-3032. <https://doi.org/10.1109/CVPR.2013.389>
- [16] Gartner, I. (2011). *Big Data: The Next Frontier for Innovation, Competition, and Productivity*. McKinsey Global Institute. [17]
- [17] Gandomi, A., & Haider, M. (2015). Beyond the hype: Big data concepts, methods, and analytics. *International Journal of Information Management*, 35(2), 137-144. <https://doi.org/10.1016/j.ijinfomgt.2014.10.007>
- [18] Zhao, W., Chellappa, R., Phillips, P. J., & Rosenfeld, A. (2003). Face recognition: A literature survey. *ACM Computing Surveys*, 35(4), 399-458. <https://doi.org/10.1145/954339.954342>
- [19] Buolamwini, J., & Gebru, T. (2018). Gender shades: Intersectional accuracy disparities in commercial gender classification. *Proceedings of the Conference on Fairness, Accountability, and Transparency*, 77-91.

- [20] Scherer, K. R., Schorr, A., & Johnstone, T. (Eds.). (2010). *Appraisal processes in emotion: Theory, methods, research*. Oxford University Press.
- [21] Simonyan, K., & Zisserman, A. (2014). *Very deep convolutional networks for large-scale image recognition*. arXiv preprint arXiv:1409.1556.
- [22] Kingma, D. P., & Ba, J. (2014). *Adam: A method for stochastic optimization*. arXiv preprint arXiv:1412.6980.
- [23] Parkhi, O. M., Vedaldi, A., & Zisserman, A. (2015). *Deep face recognition*. In *Proceedings of the British Machine Vision Conference* (pp. 41.1-41.12).
- [24] Schroff, F., Kalenichenko, D., & Philbin, J. (2015). *FaceNet: A unified embedding for face recognition and clustering*. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 815-823).
- [25] Dyer, E. L., Lu, P. J., & Perraudin, N. (2020). *Detection and prevention of face-covering in crowds using thermal and RGB imagery*. arXiv preprint arXiv:2005.08714.
- [26] Viola, P., & Jones, M. J. (2001). *Rapid object detection using a boosted cascade of simple features*. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 511-518).
- [27] Ojala, T., Pietikäinen, M., & Harwood, D. (2002). *A comparative study of texture measures with classification based on feature distributions*. *Pattern Recognition*, 29(1), 51-59.
- [28] Ahonen, T., Hadid, A., & Pietikäinen, M. (2004). *Face description with local binary patterns: Application to face recognition*. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(12), 2037-2041.
- [29] Ren, S., He, K., Girshick, R., & Sun, J. (2015). *Faster R-CNN: Towards real-time object detection with region proposal networks*. In *Advances in Neural Information Processing Systems* (pp. 91-99).

- [30] Sun, Y., Wang, X., & Tang, X. (2014). *Deep learning face representation by joint identification-verification*. In *Advances in Neural Information Processing Systems* (pp. 1988-1996).
- [31] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). *You only look once: Unified, real-time object detection*. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 779-788).
- [32] Damer, N., Grebe, J. H., & Kuijper, A. (2020). *Impact of face masks on facial recognition systems*. *IEEE International Joint Conference on Biometrics* (pp. 1-9).
- [33] Dalal, N., & Triggs, B. (2005). *Histograms of oriented gradients for human detection*. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 886-893).
- [34] Cao, Z., Chen, Z., & Zhang, Y. (2020). *FastMaskNet: Efficient detection of face masks from video streams*. In *Proceedings of the IEEE International Conference on Robotics and Automation* (pp. 9851-9857).
- [35] Taigman, Y., Yang, M., Ranzato, M. A., & Wolf, L. (2014). *DeepFace: Closing the gap to human-level performance in face verification*. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1701-1708).
- [36] Deng, J., Guo, J., Xue, N., & Zafeiriou, S. (2019). *ArcFace: Additive angular margin loss for deep face recognition*. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 4690-4699).
- [38] Sublime, J. (2022). *L'apprentissage non-supervisé et ses contradictions*. 1024 : Bulletin de la Société Informatique de France
- [39] Lienhart, R., & Maydt, J. (2002). *An extended set of Haar-like features for rapid object detection*. In *Proceedings of the International Conference on Image Processing* (Vol. 1, pp. 900-903). IEEE.

- [40] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016). SSD: Single shot multibox detector. In *European Conference on Computer Vision* (pp. 21-37). Springer, Cham.
- [41] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L. C. (2018). MobileNetV2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 4510-4520).
- [42] Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., ... & Adam, H. (2017). MobileNets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861.
- [43] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 770-778).
- [44] Glorot, X., & Bengio, Y. (2011). Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the International Conference on Artificial Intelligence and Statistics* (pp. 249-256). JMLR.
- [45] Bishop, C. M. (2006). Pattern recognition and machine learning. Springer.
- [46] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1), 1929-1958.