

الجمهورية الجزائرية الديمقراطية الشعبية

وزارة التعليم العالي والبحث العلمي



جامعة سعيدة د. مولاي طاهر

كلية التكنولوجيا

قسم : الإعلام الآلي

Mémoire de Master

Spécialité : Sécurité Informatique et Cryptographie

Reconnaissance Faciale 2D

Présenté par :

BOUSBIA Rania

BARKAT Mokhtaria

Dirigé par :

Mme.DIB Soumia



Promotion 2021 - 2022

Dédicaces

“

Avec l'expression de ma reconnaissance, je dédie ce modeste travail à mon cher et respectueux père.

A mon adorable mère.

A mes belles sœurs HAYAT, IMENE, IKRAME.

à mon frère AMINE. Que Dieu leur donne une longue et joyeuse vie.

A tous les amis qui m'encouragent. Sans oublier mon binôme BOUSBIA RANIA pour son soutien moral, sa patience et sa compréhension tout au long de ce projet.

”

Mokhtaria

Dédicaces

“

*Je dédie cette thèse à mes parents aimants, dont les paroles
d'encouragement et de pression pour la ténacité
sonnent à mes oreilles.*

*A ma famille et à de nombreux amis un sentiment particulier
de gratitude.*

*A mes sœurs WAFAA, KHAOULA n'ont jamais quitté mon
côté et sont très spéciales, elles font partie
de mon âme.*

”

Rania

Remerciements

Tout d'abord, nous remercions Dieu tout-puissant de nous avoir donné la force et la patience de compléter cette recherche.

Cette étude est dédiée de tout cœur à nos parents bien-aimés, qui nous ont inspirés et nous ont donné de la force lorsque nous avons pensé à nous abandonner, en nous apportant un soutien spirituel, émotionnel et financier.

À nos sœurs, enseignants, amis et collègues qui ont partagé des conseils et des encouragements pour terminer cette étude.

J'aimerais remercier m'encadrant, Mme Dib Soumia, pour toutes les instructions. De plus, nous tenons à nous remercier sincèrement de notre grande patience et de nos efforts.

Résumé

La reconnaissance faciale fait partie des technologies biométriques, elle est basée sur l'intelligence artificielle. Avec l'augmentation de volume des données sur le Web, le souci de sécuriser ses informations devient de plus en plus important, d'où l'apparition de plusieurs techniques pour reconnaître les visages.

Dans ce travail on s'intéresse à la reconnaissance faciale 2D, où un système de reconnaissance de visages est mis en œuvre, en utilisant la technologie de (CNN). En ce sens, la première étape consiste à détecter la présence ou l'absence d'un visage dans l'image. Puis extraire les traits du visage et les stocker dans la base de données afin que nous puissions les reconnaître plus tard. Les résultats que nous avons obtenus sont encourageants.

Mot Clé : Reconnaissance faciale 2D, deep learning, convolutional neural networks, openCV.

Abstract

Facial recognition is one of the biometric technologies, it's based on artificial intelligence. With the increase in the volume of data on the Web, the concern to secure its information becomes more and more important, hence the appearance of several techniques to recognize faces.

This work focuses on 2D facial recognition, where a face recognition system is implemented, using (CNN) technology. In this sense, the first step is to detect the presence or absence of a face in the image. Then extract the facial features and store them in the database so we can recognize them later. The results we have achieved are encouraging.

Keyword: 2D face recognition, deep learning, convolutional neural networks, openCV.

تلخيص

التعرف على الوجه هو أحد التقنيات البيومترية، فهو يعتمد على الذكاء الاصطناعي. ومع ازدياد حجم البيانات على الشبكة العالمية، يصبح الاهتمام بتأمين معلوماتها أكثر أهمية، ومن ثم ظهور عدة تقنيات للتعرف على الوجوه.

يركز هذا العمل على التعرف على الوجه ثنائي الأبعاد، حيث يتم تنفيذ نظام التعرف على الوجوه، باستخدام تقنية (CNN). وبهذا المعنى، فإن الخطوة الأولى هي اكتشاف وجود أو عدم وجود وجه في الصورة. ثم استخراج ملامح الوجه وتخزينها في قاعدة البيانات حتى نتمكن من التعرف عليها لاحقًا. والنتائج التي حققناها مشجعة.

الكلمات المفتاحية: مسح الوجه ثنائي الأبعاد، التعلم العميق، الشبكات العصبية التلافيفية، السيرة الذاتية المفتوحة.

Table des matières

Dédicaces 1	1
Dédicaces 2	2
Remerciements	3
Résumé	4
Abstract	5
ملخص.....	6
Tableaux des Matières	7
Liste des Figures	11
Abréviation.....	13
Introduction générale.....	14
I Reconnaissance faciale.....	17
I.1 Introduction	18
I.2 Les différents techniques biométriques.....	18
I.3 Définition du mot reconnaissance faciale	20
I.4 Historique de la technologie de reconnaissance faciale.....	20
I.5 Pourquoi choisir le visage.....	21
I.6 Etapes de la reconnaissance de facial.....	22
I.6.1 Le monde physique.....	22
I.6.2 Acquisition.....	23
I.6.3 Détection de visage.....	23
I.6.4 Le prétraitement.....	24
I.6.5 Extraction des paramètres.....	24
I.6.6 Classification.....	25

I.6.7 Apprentissage.....	25
I.6.8 La base des données.....	26
I.6.9 Décision.....	26
I.7 Authentification et Identification.....	27
I.8 Les techniques de reconnaissance de visage.....	27
I.8.1 Méthodes globales.....	28
I.8.2 Méthodes locales.....	30
I.8.3 Méthodes hybrides.....	30
I.9 Principales difficultés de la reconnaissance de visage.....	31
I.9.1 Changement d'illumination.....	31
I.9.2 Variation de pose.....	31
I.9.3 Expressions faciales.....	32
I.9.4 Présence ou absence des composants structurels	33
I.9.5 Les occultations	33
I.10 Applications	33
I.11 Avantages et inconvénients de la reconnaissance faciale	34
I.12 Conclusion	35
II Deep Learning.....	36
II.1 Introduction.....	37
II.2 L'intelligence artificielle (AI).....	37
II.2.1 Comprendre l'intelligence artificielle.....	38
II.3 Réseaux de neurones.....	38
II.3.1 Définition	38
II.3.2 Historique.....	38
II.3.3 Topologie.....	39
II.4 L'apprentissage en profondeur (deep Learning).....	39
II.4.1 Définition	40
II.4.2 Historique.....	41
II.4.3 Pourquoi le choix deep Learning.....	42
II.4.4 Les différentes Architectures du Deep Learning.....	43
II.4.4.1 Les réseaux de neurones convolutifs	43
II.4.4.2 Réseau de neurones récurrents.....	44
II.4.4.3 Modèle génératif.....	44

II.4.4.4 Exemples d'application de Deep Learning.....	45
II.5 Convolutional Neural Networks, CNN.....	46
II.5.1 Architecture du système.....	46
II.5.2 Les différentes couches de CNN.....	48
II.5.2.1 L'étape de convolution	48
II.5.2.1.1 Les différentes convolutions.....	49
II.5.2.1.2 Fonctions d'activation	50
II.5.2.2 L'étape de pooling.....	50
II.5.2.2.1 La mise à plat	53
II.5.2.3 La couche totalement connectée.....	53
II.4.5 Optimisation pour l'apprentissage en Deep Learning.....	54
II.6.1 Algorithmes d'optimisation de la descente de gradient.....	54
II.7 Conclusion.....	55
III L'implémentation.....	56
III.1 Introduction.....	57
III.2 Environnement de développ.....	57
III.2.1 Visual Studio Code.....	57
III.2.2 Comment mettre son Visual Studio Code ?	57
III.2.3 Pourquoi utiliser Visual Studio Code ?	57
III.2.4 Python	58
III.2.5 À quoi sert Visual Studio Code	58
III.3 Virtual Environnement	59
III.3.1 Pourquoi utiliser environnement virtuel Python	60
III.3.2 Comment activer son environnement virtuel Pytho.....	60
III.3.3 Création d'environnements virtuels	60
III.4 Bibliothèques utilisé	63
III.4.1 Face recognition	64
III.4.2 Open CV (Open Source Computer Vision Library)	68
III.4.3 NumPy	68
III.5 TensorFlow	69
III.5.1 Comment fonctionne TensorFlow	69
III.5.2 Installation de TensorFlow.....	69
III.6 Haar Cascade	70
III.7 Matplotlib	70

III.8 Keras Keras	70
III.9 La phase apprentissage	70
III.10 La Phase de tests unitaire	73
III.11 La phase analyse de résultat	75
III.12 Le code	77
III.13 Conclusion	79
Conclusion générale	80
Référence.....	81

La Liste de Figures

Fig1.1.L’empreinte digitale.

Fig1.2.L’iris.

Fig1.3.La géométrie de la main.

Fig1.7.La signature.

Fig1.8.La reconnaissance vocale.

Fig.1.9. Processus d’un système de reconnaissance de visage.

Fig.1.10. Exemple d’acquisition d’une image.

Fig.1.11. Détection de visage.

Fig.1.12. Phase d'apprentissage.

Fig1.13. Architecture d’un système biométrique en mode identification.

Fig.1.14. Phase d'identification/authentification.

Fig.1.14. Techniques de reconnaissance de visage.

Fig.1.15. Les étapes d'une approche globale de reconnaissance faciale.

Fig.1.16. Exemples de changement d’illumination.

Fig.1.17. Exemples de variation de pose.

Fig.1.18. Exemples de variation d’expressions.

Fig.1.19. Exemples de composants structurels.

Figure 2.1 La relation entre l’intelligence artificielle, le ML et le deep Learning.

Figure 2.2 Schéma illustratif de DL avec plusieurs couches.

Figure 2.3 Comparaison entre la machine Learning et le Deep Learning.

Figure 2.4 Différents modèles du Deep Learning.

Figure 2.5 architecture générale d’un CNN.

Figure 2.6 Architecture proposée.

Figure 2.7 Phase de convolution.

Figure 2.8 différents types de convolutions.

Figure 2.9 Exemple de description de la couche pooling.

Figure 2.10 Calcul du pooling sur une image 4×4.

Figure 2.11 Mise à plat des images.

Figure 2.12 Mise à plat des images.

Figure 3.1 Le logo de python.

Figure 3.2 création d'environnements virtuels.

Figure 3.3 création environnement virtuel de python.

Figure 3.4 les bibliothèques.

Figure 3.5 Input.

Figure 3.6 Out put.

Figure 3.7 Input.

Figure 3.8 output.

Figure 3.9 Input.

Figure 3.10 Output.

Figure 3.11 photo de lamamra.

Figure 3.12 photo de tebboune.

Figure 3.13 Capture de code de la phase de teste.

Figure 3.14 Captures de la base de teste.

Figure 3.15 capture de la reconnaissance (Tebboune) .

Figure 3.16 capture de la reconnaissance (Lamamra) .

Figure 3.17 Capture de code.

Figure 3.18 Capture de code.

Abréviation

RF2D: Reconnaissance Facial 2 Dimension.

LDA: linear discriminant analysis.

PCA: Principal Component Analysis.

ICP: Iterative Closest Point.

AI : Artificial Intelligence.

DL: Deep Learning.

ML: Machine Learning.

RN : Réseaux de neurones

RNN: Recurrent Neural Network.

CNN: Convolutional Neural Network.

ANN: Artificial Neural Network.

DCNN: Deep Convolutional Neural Networks.

SVM: Support vector machine.

ReLU : Rectified Linear Unit.

Introduction générale

Contexte

La sécurité des personnes, les propriétés et de l'information doit être garantie, ce qui est en fait l'une des principales préoccupations de nos sociétés, surtout après la grande propagation du terrorisme dans le monde. En effet, les personnes désireuses de franchir les frontières doivent prouver leur identité à l'aide de leur passeport, les personnes désireuses de traverser les bâtiments ou les établissements d'enseignement doivent valider leur carte d'accès, les personnes désirant accéder aux services bancaires doivent se connecter à l'aide d'un identifiant et d'un mot de passe. Néanmoins, ces méthodes traditionnelles présentent de grandes faiblesses pour la vérification de l'identité. En effet, l'identité d'une personne est directement liée à ce qu'elle possède (comme un passeport, une carte d'accès, etc.) ou qu'elle connaît (mot de passe, code PIN, etc.). Néanmoins, les codes PIN et les mots de passe peuvent être oubliés ou compromis et les cartes d'accès peuvent être falsifiées ou dupliquées, ce qui entraîne une éponge d'identité. À cet égard, les experts recherchent une technologie qui résout ces problèmes en donnant plus de commodité aux personnes et en assurant un accès hautement sécurisé, en reliant l'identité d'une personne à ce qu'elle est et non à ce qu'elle possède ou sait. La biométrie est la technologie la plus appropriée pour la vérification de l'identité et/ou l'identification des personnes en utilisant leurs caractéristiques physiologiques, y compris les caractéristiques biologiques, morphologiques et comportementales. Cette technologie rend le vol de données d'identité plus difficile et, par conséquent, accroît la confiance des utilisateurs, car la présence physique est nécessaire pendant l'identification.

Dans notre travail, nous avons choisi la reconnaissance faciale comme moyen d'identification par rapport à d'autres méthodes parce que cette identification est naturellement utilisée par un être humain, donc ce type de reconnaissance ne s'arrête pas à l'identification du visage, mais peut s'appliquer à l'emplacement d'une personne dans une foule, contrairement à d'autres méthodes, et ne nécessite pas d'équipement d'acquisition très complexe, c'est-à-dire qu'une simple photo peut acquérir la forme du visage d'une personne, puis supprimer certaines caractéristiques. Les caractéristiques essentielles pour la reconnaissance faciale sont les yeux, la bouche, le visage, le nez, etc. Selon le système utilisé, la personne doit être positionnée devant la caméra pour capturer leur visage. Les données biométriques obtenues sont comparées au fichier de référence.

Cette thèse porte sur un sujet d'identification. Un système d'identification vise à répondre à la question : « Qui est cette personne? » Il faut comparer les données biométriques à celles de la base de données. Donc il est responsable de la découverte de l'identité d'une personne inconnue dans un ensemble de données. Plusieurs méthodes ont été développées dans la littérature pour la reconnaissance faciale. Dans notre travail, nous avons opté pour une technique basée sur les réseaux neuronaux appelés Convolutional Neural Networks (CNN) qui est un type de réseau neuronal avec apprentissage profond, ou Deep Neural Network. Ce dernier comporte plusieurs couches cachées. CNN se compose de deux parties très distinctes, une partie de l'extraction qui peut être utilisée pour simplifier une image d'entrée, réduire sa taille, et une partie de la classification qui classe ces données.

Organisation du mémoire

La suite de ce mémoire s'organise de la manière suivante :

Chapitre 01 : “Reconnaissance facial 2D”

Dans ce chapitre, nous essayons de fournir un aperçu de la technologie biométrique et d'introduire la technologie de reconnaissance faciale et de comprendre ses propriétés, après nous avons donné un aperçu de l'histoire de cette technique et discuté des différentes étapes pour identifier le visage. Après cela, nous introduirons des techniques de reconnaissance faciale, ensuite illustrent les principales difficultés qui entravent la reconnaissance faciale et les diverses applications qui s'appliquent à la vie quotidienne. Pour conclure, un tableau résumant les avantages et les inconvénients est dressé. Enfin, nous terminerons ce chapitre par une conclusion qui énumère les points étudiés.

Chapitre 02 : “Deep Learning”

La solution de ces problèmes intuitifs est traitée dans ce chapitre. Cette solution est de permettre aux ordinateurs d'apprendre de l'expérience et de comprendre le monde en termes de hiérarchie des concepts, chaque concept défini en termes de sa relation avec des concepts plus simples. En rassemblant les connaissances de l'expérience, cette approche permet d'éviter la nécessité pour les opérateurs humains de préciser formellement toutes les connaissances que la machine les besoins. La hiérarchie des concepts permet à l'ordinateur d'apprendre des concepts complexes en les construisant à partir de concepts plus simples. Si nous dessinons un graphique montrant comment ces concepts sont construits les uns sur les autres, le graphique est profond, avec de nombreuses couches. Pour cette raison, nous

appelons cette approche de l'apprentissage profond

Chapitre 03 : “L'implémentation”

Ce chapitre est composé de deux parties, l'implémentation du système et les résultats expérimentaux des tests. il présenter les étapes de l'implémentation de l'approche proposée dans le cadre d'un système de reconnaissance faciale de visage. Nous commençons tout d'abord par la présentation des ressources, du langage et de l'environnement de développement que nous avons utilisé. Puis les étapes de la réalisation du modèle et on termine par les tests effectués.

Enfin, la conclut générale de ce mémoire.

Chapitre I

Reconnaissance

Faciale

1. Introduction :

La biométrie est devenue à la mode dans les domaines qui requiert un niveau élevé de sécurité et de contrôle. Et parmi toutes les technologies biométriques qui existent, la reconnaissance des visages est l'une des technologies les plus utilisées et les plus adaptées.

Au cours des vingt dernières années, La reconnaissance de visage a été abordée par plusieurs chercheurs et chaque année des avancements dans ce domaine ont vu le jour. Où la reconnaissance automatique des visages est devenue un enjeu primordial, notamment dans les domaines de sécurité, ceci est dû aux besoins du monde actuel et aussi à ses caractéristiques avantageuses.

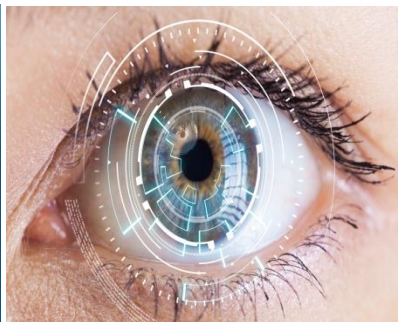
Plusieurs méthodes de reconnaissance faciale ont été proposées sur deux axes principaux: la reconnaissance d'images statiques et l'identification de séquences d'images (vidéo). Sans aucun doute, les investissements dans la reconnaissance faciale conduisent à une diversité dans les domaines d'application (haute sécurité, télésurveillance, contrôle d'accès ...).

Dans ce chapitre, nous donnerons un aperçu des technologies biométriques utilisées dans le domaine de la sécurité, Ensuite, nous discuterons de la généralité de la technologie de reconnaissance faciale et de comprendre ses caractéristiques.

2. Les différents techniques biométriques:

La biométrie consiste à identifier une personne à partir d'une ou de plusieurs caractéristiques physiologiques (empreintes digitales, visage, iris, contour de la main, etc.), ou comportementales (signature, démarche, etc.). Une autre définition de la biométrie est donnée par Roethenbaugh [1]: « La biométrie s'applique à des particularités ou des caractères humains uniques en leur genre et mesurables, permettant de reconnaître ou de vérifier automatiquement l'identité ». Il existe plusieurs caractéristiques physiques uniques pour un individu, ce qui explique la diversité des systèmes appliquant la biométrie, selon que l'on prend en compte :

- L'empreinte digitale
- La géométrie de la main
- L'iris
- La rétine
- Le visage
- La dynamique de frappe au clavier
- La reconnaissance vocale
- La dynamique des signatures
- L'ADN...

**Fig1.1.**L'empreinte digitale [w1]**Fig1.2.**L'iris [w2].**Fig1.3.**La géométrie de la main [w3].**Fig1.4.** Mote de passe [w4]**Fig1.5.**Le visage [w5]**Fig1.6.**ADN [w6]**Fig1.7.**La signature [w7].**Fig1.8.**La reconnaissance vocale [w8].

3. Définition du mot reconnaissance faciale :

La reconnaissance faciale est un moyen d'identifier ou de confirmer l'identité d'un individu grâce à son visage. Les systèmes de reconnaissance faciale peuvent servir à l'identification de personnes sur des photos, dans des vidéos ou en temps réel.

La reconnaissance faciale est une catégorie de sécurité biométrique. Cette technologie est surtout utilisée pour la sécurité et l'application de la loi, afin d'identifier des criminels ou des disparus et autoriser l'accès à certains lieux publics. Bien que d'autres domaines s'y intéressent de plus en plus, les logiciels de reconnaissance faciale sont créés pour aider à l'utilisation d'applications telles que les réseaux sociaux qui reconnaissent des visages sur les photos mises en ligne ou le déverrouillage d'appareils. Analysant les particularités d'un visage, l'algorithme compare ses résultats à une base de données établie et peut être utilisé en marketing pour analyser et cibler une clientèle selon son genre ou son âge [w9].

4. Historique de la technologie de reconnaissance faciale :

1964 : Des chercheurs américains, Woodrow Bledsoe, Helen Chan et Charles Bisson, étudient la programmation d'ordinateurs en vue de reconnaître des visages. Ils imaginent une méthode semiautomatique : des opérateurs doivent rentrer dans l'ordinateur une vingtaine de mesures, comme la taille de la bouche ou des yeux [w10].

1991 : Premier exemple réussi de technologie de reconnaissance faciale, Eigenfaces, qui utilise la méthode statistique d'analyse en composantes principales, est présenté par Alex Pentland et Matthew Turk, du Massachusetts Institute of Technology [w10].

2011 : Tout s'accélère grâce au deep learning, une méthode d'apprentissage automatique s'appuyant sur des réseaux de neurones artificiels. C'est l'ordinateur qui sélectionne les points à comparer : plus il est alimenté en images, mieux il apprend [w10].

2012 : Un premier outil français de comparaison d'images de visages est testé dans le nouveau fichier du traitement des antécédents judiciaires (TAJ) [w10].

2014 : Facebook sait identifier votre visage grâce à son algorithme maison, Deepface. Le réseau social affirme que sa méthode, avec ses 97 % d'efficacité, tutoie les performances de l'œil humain [w10].

2018 : La police chinoise affirme avoir arrêté un suspect grâce à l'utilisation, en direct, de la reconnaissance faciale. L'homme, soupçonné de "crime économique", avait été repéré dans un concert de la pop star Jacky Cheung à Nanchang, dans la province du Jiangxi. Son visage, répertorié dans une base de données nationale, a été repéré dans une foule de 50 000 personnes [w10].

2019 : En Chine, les personnes souhaitant acheter un nouveau téléphone doivent désormais accepter que l'opérateur numérise leur visage [w10].

5. Pourquoi choisir le visage ?

La reconnaissance de visages est la technique la plus commune et populaire. Elle reste la plus acceptable puisqu'elle correspond à ce que les humains utilisent dans l'interaction visuelle; et par rapport aux autres méthodes, la reconnaissance du visage s'avère plus avantageuse, d'une part c'est une méthode non intrusive, c'est-à-dire elle n'exige pas la coopération du sujet (en observant les individus à distance), et d'une autre part les capteurs utilisés sont peu coûteux (une simple caméra) contrairement à l'empreinte digitale et l'iris où le sujet devra être très proche du capteur et devra coopérer pour l'acquisition de l'image sans oublier le coût de l'équipement nécessaire pour l'acquisition (équipement spécial coûteux). Malgré que certains disent que la reconnaissance de visages est une biométrie relativement peu sûre, sur le fait que le signal acquis est sujet à des variations beaucoup plus élevées que d'autres caractéristiques, comme la variation de l'éclairage, le changement de la position du visage, la présence ou l'absence de lunettes et autres; mais, au cours de ces dernières années plusieurs techniques de traitements d'images sont apparues, telle que la détection du visage, la normalisation de l'éclairage, etc. Sans oublier le développement considérable des technologies des caméras numériques, ce qui néglige l'effet de ces problèmes [w11].

6. Etapes de la reconnaissance de faciale :

La reconnaissance faciale est un système permettant d'identifier et de confirmer les personnes en contrôlant si celles-ci appartiennent à la base de données du système. L'image suit un processus de reconnaissance faciale spécifique contenant plusieurs étapes qui peuvent être illustrées dans le diagramme de la figure 1.9 ci-dessous:

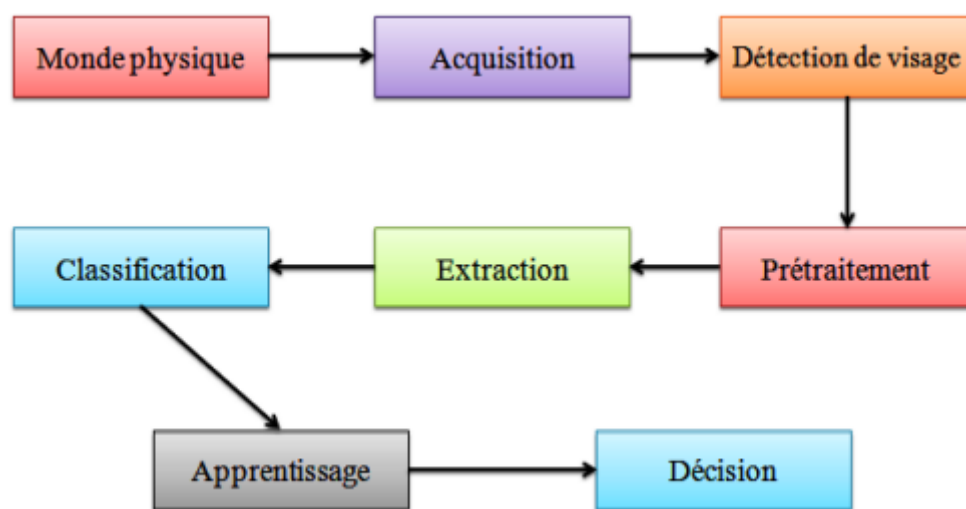


Fig.1.9: Processus d'un système de reconnaissance de visage [2].

6.1 Le monde physique :

C'est le monde réel en dehors du système avant l'acquisition de l'image. Dans cette étape, on tient compte généralement de trois paramètres essentiels: L'éclairage, la variation de posture et l'échelle. La variation de l'un de ces trois paramètres peut conduire à une distance entre deux images du même individu, supérieure à celle séparant deux images de deux individus différents, et par conséquent une fausse identification [3].

6.2 Acquisition :

Le système d'acquisition est généralement équipé d'un capteur qui permet aux utilisateurs d'obtenir une fonction spécifique (par exemple, une camera pour capter une photo, etc.). Un appareil photo nous permet d'avoir une image 2D du visage à partir d'une scène 3D comme indiqué dans la Figure 1.10.

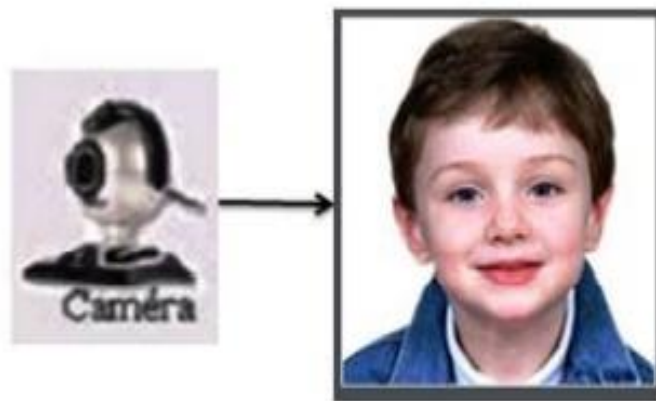


Fig.1.10: Exemple d'acquisition d'une image [2].

6.3 Détection de visage :

La détection de visage est une étape très intéressante dans le domaine de reconnaissances de visage [4]. Plusieurs travaux de recherches ont été effectués dans ce domaine. Ils ont donné lieu au développement d'une multitude de techniques allant de la simple détection du visage, à la localisation précise des régions caractéristiques du visage, tels que les yeux, le nez, les sourcils, la bouche, les lèvres, les oreilles, etc [5].

Un visage est considéré correctement détecté si la taille d'image extraite ne dépasse pas 20% de la taille réelle de la région faciale [6] comme illustré par la figure 1.11. Cette étape peut faire la détection de la couleur de peau, la forme de la tête et il existe plusieurs méthodes détectant les différentes caractéristiques du visage.



Fig.1.11: Détection de visage [2].

6.4 Le prétraitement :

Les données délivrées par les capteurs primaires ne sont qu'une représentation initiale de celles-ci d'où la nécessité d'un traitement antérieur. L'image brute peut être affectée par divers facteurs provoquant sa dégradation, pouvant être bruyante, c'est-à-dire contenir de fausses informations dues à des dispositifs optiques ou électroniques. Le rôle de cette étape est d'éliminer les parasites accompagnants l'image, provoqués par la qualité de ces dispositifs.

Ceci est nécessaire car l'image ne peut jamais être sans bruit car le fond et la lumière sont généralement inconnus. Il existe plusieurs types de traitement et d'optimisation de la qualité d'image, tels que la normalisation, les graphiques, le filtrage, la correction gamma ou des méthodes plus complexes telles que le lissage anisotrope [2].

6.5 Extraction des paramètres :

L'extraction des paramètres est le cœur du système de reconnaissance qui extrait les informations d'image qui seront stockées dans la mémoire pour une utilisation ultérieure dans l'étape de décision. Le choix de cette information utile réside dans la création d'un modèle de visage, qui doit être discriminatoire. Cette analyse est appelée propriétés d'indexation, de représentation, de modélisation ou d'extraction. L'efficacité de cette étape a un impact direct sur la performance du système de reconnaissance faciale [3].

Voici quelques façons d'extraire des informations pertinentes:

- **Méthodes globales:** elles utilisent toute la surface du visage comme entrée dans le système de reconnaissance. L'une des méthodes les plus largement utilisées pour la représentation faciale dans son ensemble est la PCA.
- **Méthodes locales:** Ces méthodes dérivent spécifiquement les caractéristiques faciales locales telles que les yeux, le nez et la bouche, puis utilisent la géométrie et / ou l'apparence comme une entrée pour la classe.

6.6 Classification :

Lorsque les formulaires sont stockés dans la base de données, le système se compose d'échantillons similaires de nombreuses personnes sélectionnées ainsi que d'une liste limitée de candidats. Cette étape consiste à modéliser les paramètres extraits de la ou des faces de chaque individu en fonction de leurs caractéristiques communes. Un modèle est une collection d'informations utiles, uniques et non récurrentes qui identifie une ou plusieurs personnes ayant des similitudes [2].

6.7 Apprentissage :

L'apprentissage consiste à retenir les modèles calculés pendant la phase d'analyse des personnes connues. Ce modèle est une représentation intégrée d'images pour faciliter l'identification, mais aussi la quantité de données stockées sous une forme ou une autre. Cette étape correspond aux références interactives réelles qui seront enregistrées dans la base de données comme le montre la figure 1.12 suivante :

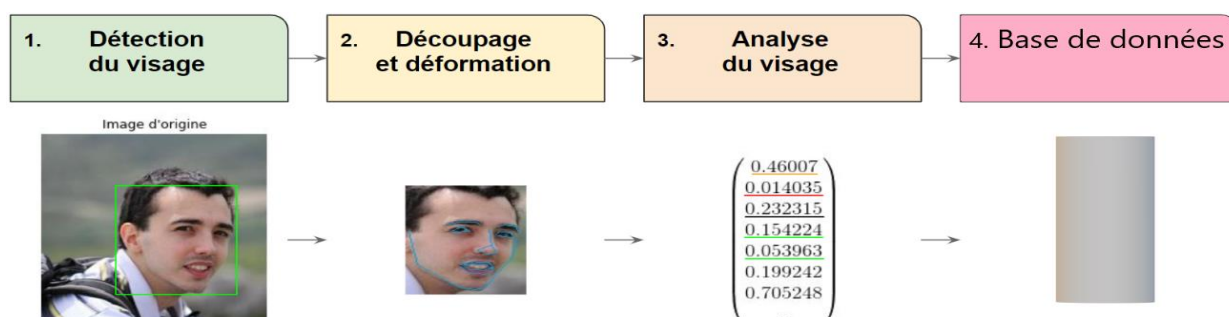


Fig.1.12: Phase d'apprentissage [w12].

6.8 La base des données :

La base de données (Data base en anglais) est un centre informatique pour la collection, la coordination, le stockage et l'utilisation d'informations. Il peut stocker et récupérer des données brutes ou des informations sur le sujet en activité [2]. Dans la grande majorité des cas, cette information est très organisée, toutes les données sont situées au même endroit et sur le même support.

La création d'une base de données pour les systèmes de reconnaissance faciale est une tâche très complexe car il existe de nombreux facteurs pour obtenir les données et exiger une application stricte du protocole afin que les images des individus obtenus puissent être comparées entre elles.

6.9 Décision :

La décision fait partie du système dans lequel nous décidons si l'individu appartient à tous les visages ou non. Dans cette phase, le système d'identification consiste à trouver le modèle correspondant au visage pris à partir de ceux stockés dans la base de données (voir la figure 1.13 ci-dessous), dans ce cas quelle est son identité. Par conséquent, la résolution est l'aboutissement de ce processus. Il peut être évalué au taux de reconnaissance (fiabilité) déterminé par le taux de résolution de la décision.

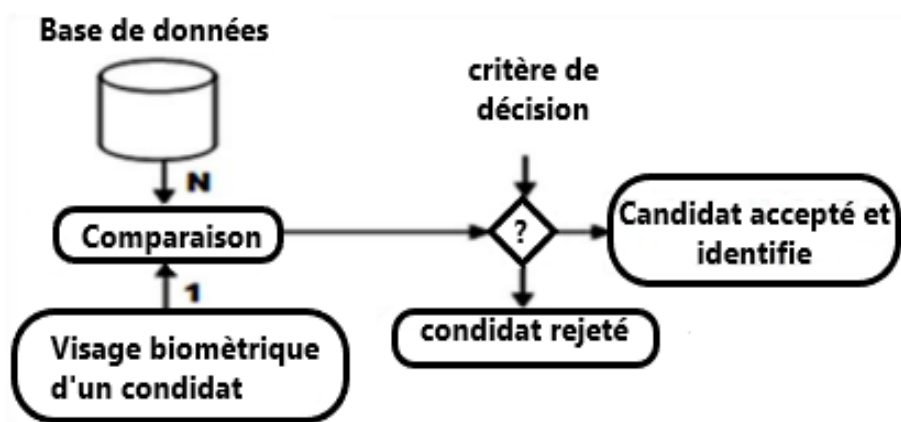


Fig1.13: Architecture d'un système biométrique en mode identification [2].

7. Authentification et Identification :

Le processus d'identification de la comparaison des caractéristiques physiologiques et comportementales implique des modèles stockés dans une base de données, dans laquelle un utilisateur encore inconnu en tant que biologiste ou très similaire est défini selon un modèle standard ou des éléments spécifiques. Le système peut uniquement spécifier un utilisateur déjà présent dans la base de données. Contrairement à l'identité, l'authentification implique de comparer le modèle d'entrée avec l'identité demandée. Ici, l'utilisateur peut être classé comme un vrai utilisateur ou faux, comme le montre la figure 1.14 ci-dessous.

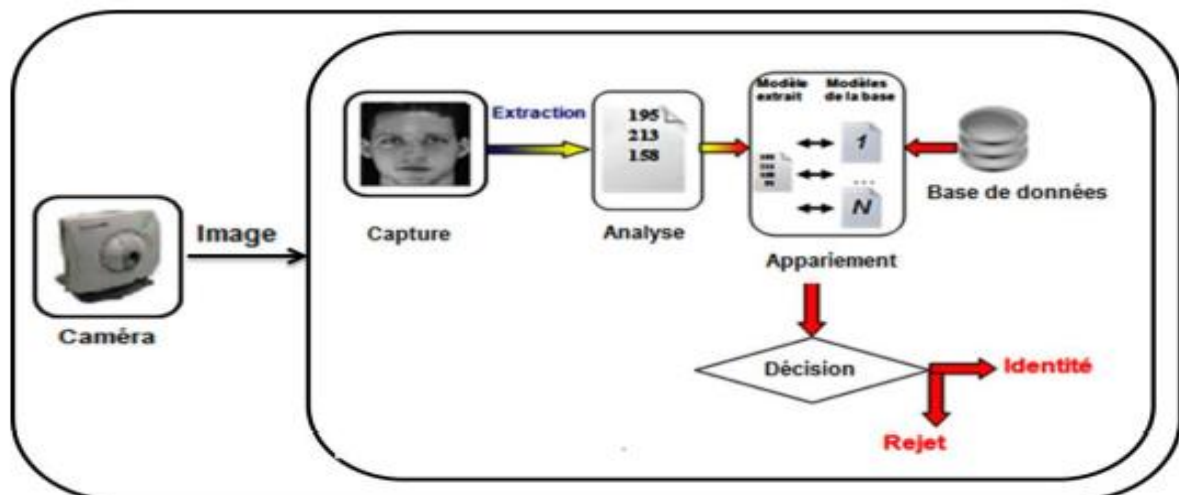


Fig.1.14: Phase d'identification/authentification [2].

8. Les techniques de reconnaissance de visage :

Les méthodes de reconnaissance faciale peuvent être divisées en trois familles principales, les méthodes globales, les méthodes locales et les méthodes hybrides, comme indiqué dans le diagramme de la figure 1.14 ci-dessous:

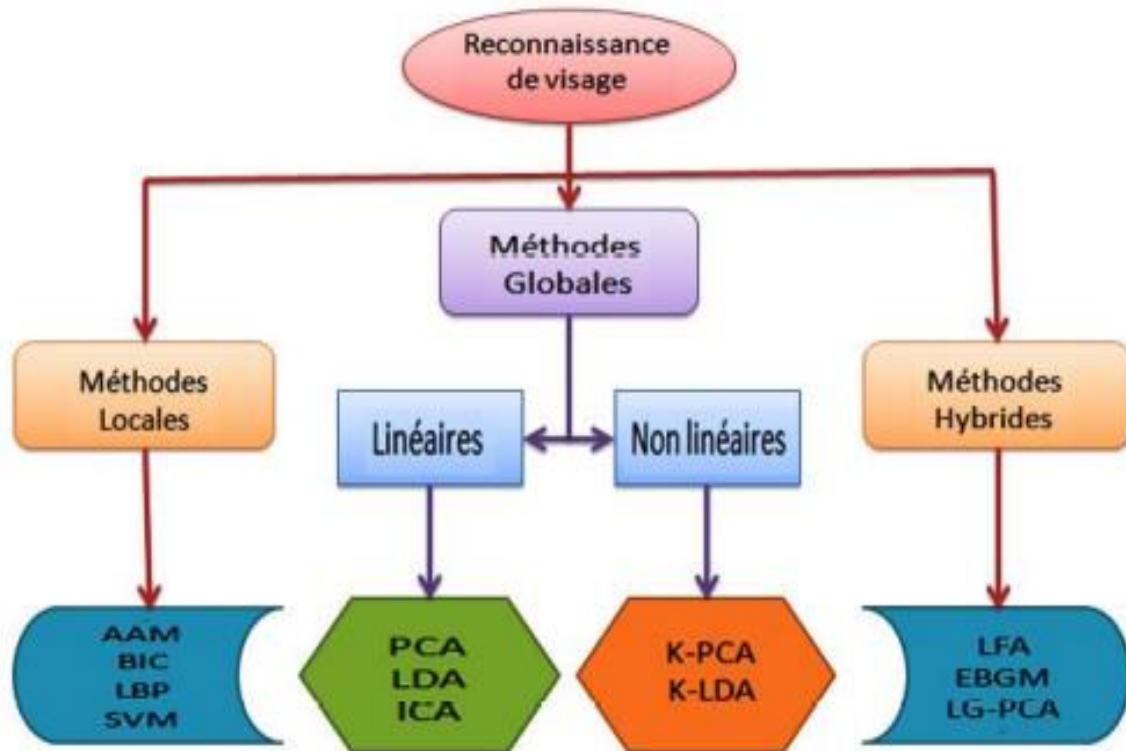


Fig.1.14: Techniques de reconnaissance de visage [2].

6.1 Méthodes globales :

Les méthodes globales sont basées sur des techniques d'analyse statistique bien connues. Il n'est pas nécessaire de repérer certains points caractéristiques du visage (comme les centres des yeux, les narines, le centre de la bouche, etc.) à part pour normaliser les images [7].

Le principe de ces méthodes est d'utiliser le visage complet comme une source d'information sans segmentation de ses parties, elles se basent principalement sur l'information de pixels. Dans ces méthodes, les images de visage (qui peuvent être vues comme des matrices des valeurs des pixels) sont traitées de manière globale et sont généralement transformées en vecteurs, plus faciles à traiter.

Ces algorithmes s'appuient sur des propriétés statistiques bien connues et utilisent l'algèbre linéaire. Ils sont relativement rapides à mettre en œuvre mais sont sensibles aux problèmes de changement d'illumination, de pose et d'expression faciale [4]. Les étapes d'une approche globale pour identifier le visage sont les étapes montrées dans la figure 1.15 ci-dessous.

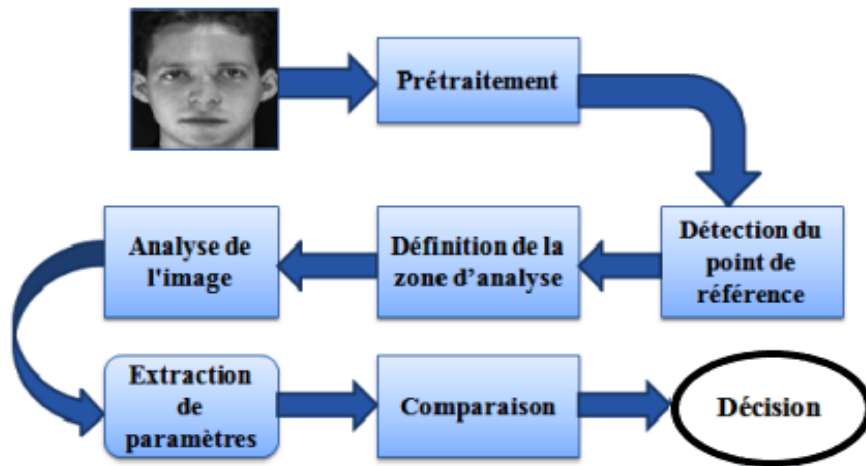


Fig.1.15: Les étapes d'une approche globale de reconnaissance faciale [2].

L'avantage des méthodes globales est qu'elles conservent implicitement les informations de texture et de forme utiles pour la reconnaissance de visage, ainsi qu'elles permettent une meilleure capture de l'aspect global du visage que les présentations locales, de plus elles sont rapides à mettre en œuvre [8].

Nous pouvons distinguer deux types de techniques parmi les méthodes globales :

- ❖ Les techniques linéaires sont des techniques qui projettent linéairement les données des visages d'espace de grande dimension sur un espace de plus faible dimension :
 - **PCA** (Principal Component Analysis).
 - **LDA** (Linear Discriminate Analysis).
 - **ICA** (Independent Component Analysis).
- ❖ Les techniques non linéaires sont basées sur la notion mathématique de noyau (Kernel) comme le :
 - **K-PCA** (Kernel Principal Composant Analysis).
 - **K-LDA** (Kernel Linear Discriminate Analysis).

Ici, une projection non linéaire (réduction de dimension) de l'espace de l'image sur l'espace de caractéristiques est effectuée; les variétés présentes dans l'espace de caractéristiques résultant deviennent simples, de même que les subtilités des variétés qui sont préservées [9]. L'inconvénient majeur des méthodes globales est l'insuffisance de l'information extraite représentant le visage, car ces méthodes ne prennent pas en considération les variations locales en essayant de concentrer le maximum d'énergie [10].

6.2 Méthodes locales :

Les méthodes locales reposent sur la reconnaissance à priori que l'on possède sur les objets caractéristiques pour pouvoir les localiser. Ces méthodes peuvent être classées en deux catégories, la première concerne les méthodes basées sur les caractéristiques locales, il s'agit des méthodes d'extraction et de localisation des points d'intérêts, la deuxième catégorie est celle des méthodes basées sur l'apparence faciale qui divisent l'image de visage en petites régions (ou patches) de caractéristiques locales [11]. Parmi ces méthodes, on peut citer :

- **AAM** (Active Apparence Model).
- **BIC** (Les Approches Bayésiennes).
- **LBP** (Local Binary Pattern).
- **SVM** (Support Vector Machines).

Toutes ces méthodes ont l'avantage de pouvoir modéliser plus facilement les variations de pose, d'illumination et d'expression par rapport aux méthodes globales. Cependant, ils sont plus lourds à utiliser car il est souvent nécessaire de placer manuellement un grand nombre de points sur le visage, alors que les méthodes globales ont seulement besoin de connaître la position des yeux pour normaliser les images, ce qui peut être fait automatiquement et assez fiable à partir d'un algorithme de détection.

6.3 Méthodes hybrides :

Les méthodes hybrides combinent les avantages des méthodes globales et locales qui combinent la découverte de caractéristiques structurelles (ou d'ingénierie) avec l'extraction de caractéristiques d'apparence locales. Qui à son tour augmente la stabilité de la performance de reconnaissance lors du changement de forme tels que l'éclairage, les expressions faciales et ainsi de suite.

La combinaison efficace des caractéristiques locales et globales reste un problème pour l'instant et il y a peu de travail sur son application au problème de la reconnaissance faciale [12]. Parmi les techniques de ces méthodes on trouve :

- **LFA** (Local Feature Analysis).
- **EBGM** (Elastic Buch Graph Matching).
- **LG-PCA** (Log Gabor PCA).

9. Principales difficultés de la reconnaissance de visage :

Pour le cerveau humain, la reconnaissance faciale est une mission visuelle de haut niveau. Bien que les gens puissent détecter des objets et les reconnaître dans une scène sans trop de problèmes, la création d'un système automatique qui effectue ces tâches est un défi sérieux. Ce défi est beaucoup plus grand que les conditions pour obtenir des images très différentes. La différence entre les sujets est limitée par la similitude physique entre les individus. D'un autre côté, la différence dans ce sujet est plus grande. Cela peut être attribué à plusieurs facteurs que nous analysons ci-dessous [2].

6.1 Changement d'illumination :

Certains facteurs tels que l'éclairage (répartition de la source de lumière, intensité, spectre) et les caractéristiques de la caméra affectent l'apparence d'un visage dans l'image acquise, comme le montre la figure 1.16 suivante :



Fig.1.16: Exemples de changement d'illumination [2].

6.2 Variation de pose :

C'est une variation de la rotation et c'est un gros problème avec les systèmes de reconnaissance faciale. En effet, de nombreux tests ont montré que la rotation de la tête n'entraîne pas de réduction significative des taux de détection à $\pm 25^\circ$. Mais si cette rotation dépasse ce seuil, cela réduira les performances. La figure 1.17 illustre une personne à laquelle les changements sont faits.



Fig.1.17: Exemples de variation de pose [w13].

6.3 Expressions faciales :

La déformation faciale due aux expressions faciales affecte principalement la partie inférieure du visage. L'information faciale trouvée en haut du visage reste presque constante, ce qui est habituellement suffisant pour mener à bien le processus d'identification. Cependant, puisque l'expression faciale modifie l'apparence du visage, elle entraîne nécessairement une diminution du taux de reconnaissance. L'identification faciale avec l'expression faciale est un problème difficile qui est toujours pertinent et reste non résolu [13], la figure 1.18 ci-dessous montre quelques expressions faciales.



Fig.1.18: Exemples de variation d'expressions [w13].

6.4 Présence ou absence des composants structurels :

Des aspects particuliers tels que la barbe, la moustache et les lunettes, comme le montre la figure 1.19, provoquent des changements importants dans les composants structuraux du visage, notamment la forme, la couleur, la taille, etc.



Fig.1.19. Exemples de composants structurels [2].

6.5 Les occultations :

Les visages peuvent être partiellement masqués par d'autres objets qui couvrent le visage. En effet, dans une image qui contient un groupe de personnes, par exemple, le visage peut masquer partiellement d'autres visages [2].

10.Applications :

Les systèmes de reconnaissance faciale sont plus applicables à la vie quotidienne. Des exemples d'applications de reconnaissance faciale comprennent:

- Les réseaux sociaux sont utilisés sur Internet pour identifier une personne à travers l'image de son visage et pour vérifier l'identité des achats en ligne, sur le smart phone pour déverrouiller, ou par des services de sécurité pour reconnaître les individus recherchés.
- Réduire la fraude par la vérification d'identité dans les banques, les élections, les soins de santé, les paiements de prestations, etc.

- Bien utilisé dans les applications de contrôle des frontières, dans la sécurité des institutions et des zones urbaines et dans l'identification des conducteurs.
- Les principaux utilisateurs de cette technique sont les policiers qui les utilisent pour des raisons de sécurité.
- La technologie de reconnaissance faciale peut être utilisée pour identifier les personnes qui ont un casier judiciaire et localiser les criminels et les trouver.
- Les éditeurs de logiciels peuvent également tirer parti de la technologie de reconnaissance faciale pour authentifier les utilisateurs et réduire les risques de pénétration, tels que Google et Facebook.
- Cette technique peut être utilisée dans les robots pour surveiller en alertant s'il y a des objets suspects dans les zones sensibles [2]. Il peut également être utilisé dans les banques et les lieux publics tels que les aéroports, les centres commerciaux, les cinémas et autres.

11. Avantages et inconvénients de la reconnaissance faciale :

Les avantages :

- Facile à mettre en œuvre et à tester car il ne nécessite pas d'équipement spécial, mais tout ce dont vous avez besoin est d'une caméra haute résolution.
- Bien accepté par le public.
- Aucune action de l'utilisateur (n'implique pas de coopération de l'utilisateur).
- Pas de contact physique.
- Technique peu coûteuse (Capteurs très bon marché) [2].

Les inconvénients :

- Sensible à l'environnement d'acquisition (éclairage, position, expression du visage...)
- Les vrais jumeaux ne sont pas différenciés.
- Sensible aux changements (barbe, moustache, lunettes, piercing, chirurgie...)

12. Conclusion :

Dans ce chapitre, nous avons présenté la technologie de reconnaissance faciale pour identifier les personnes. Nous avons également fourni un aperçu des étapes et des techniques de reconnaissance faciale (PCA, ICA, etc.). Cette étude nous a permis de savoir que la reconnaissance faciale attire davantage l'attention de la communauté scientifique car elle présente de nombreux défis et obstacles technologiques. Enfin, nous avons mis en évidence les différentes difficultés inhérentes à la reconnaissance automatique des visages, ce qui nous a permis d'identifier les problèmes qui pourraient être résolus.

Chapitre II

Deep

Learning

1. Introduction

Depuis longtemps, les inventeurs rêvent de créer des machines pensent comme l'être humaine. Aujourd'hui, l'intelligence artificielle (AI) est un domaine florissant avec de nombreuses applications pratiques et des sujets de recherche actifs. Nous nous tournons vers des logiciels intelligents pour automatiser le travail de routine, comprendre la parole ou les images, faire des diagnostics en médecine et soutenir la recherche scientifique fondamentale. Aux débuts de l'intelligence artificielle, le domaine a rapidement abordé et résolu des problèmes qui sont intellectuellement difficiles pour les êtres humains mais relativement simples pour les problèmes informatiques qui peuvent être décrits par une liste de règles mathématiques formelles. Le véritable défi à l'intelligence artificielle est de résoudre des tâches qui sont faciles pour l'humaine à effectuer, mais difficile de les décrire formellement, des problèmes que nous résolvons de façon intuitive, qui se sentent automatique, comme la reconnaissance des paroles ou des objets dans les images. La solution de ces problèmes intuitifs est traitée dans ce chapitre. Cette solution est de permettre aux ordinateurs d'apprendre de l'expérience et de comprendre le monde en termes de hiérarchie des concepts, chaque concept défini en termes de sa relation avec des concepts plus simples. En rassemblant les connaissances de l'expérience, cette approche permet d'éviter la nécessité pour les opérateurs humains de préciser formellement toutes les connaissances que la machine les besoins. La hiérarchie des concepts permet à l'ordinateur d'apprendre des concepts complexes en les construisant à partir de concepts plus simples. Si nous dessinons un graphique montrant comment ces concepts sont construits les uns sur les autres, le graphique est profond, avec de nombreuses couches. Pour cette raison, nous appelons cette approche de l'apprentissage profond

2. L'intelligence artificielle (AI)

Fait référence à la simulation de l'intelligence humaine dans les machines qui sont programmés pour penser comme les humains et imiter leurs actions. Le terme peut

également être appliqué à toute machine qui présente des traits associés à un esprit humain tel que l'apprentissage et la résolution de problèmes.

2.1 Comprendre l'intelligence artificielle

L'intelligence artificielle est basée sur le principe selon lequel l'intelligence humaine peut être définie de manière à ce qu'une machine puisse facilement l'imiter et exécuter des tâches, des plus simples à celles qui sont encore plus complexes. Les objectifs de l'intelligence artificielle comprennent l'apprentissage, le raisonnement et la perception. Avec les progrès technologiques, des critères précédents qui ont défini l'intelligence artificielle deviennent obsolètes. Par exemple, les machines qui calculent les fonctions de base ou reconnaître du texte grâce à la reconnaissance des caractères optimal ne sont plus considérés pour incarner l'intelligence artificielle, puisque cette fonction est maintenant pris pour acquis en tant que fonction informatique inhérente. L'AI évolué de façon continue au profit de nombreuses industries différentes. Les machines sont reliées selon une approche interdisciplinaire basée sur les mathématiques, l'informatique, la linguistique, la psychologie, etc.

3. Réseaux de neurones

3.1 Définition

Un réseau de neurones artificiels est un système dont la conception est à l'origine inspirée du fonctionnement des neurones biologiques, et qui par la suite s'est rapproché des méthodes statistiques [w14]. Les réseaux de neurones artificiels sont des réseaux fortement connectés par des processeurs élémentaires fonctionnant en parallèle. Chaque processeur élémentaire (neurone artificiel) calcule une sortie unique sur la base des informations qu'ils reçoivent.

3.2 Historique

Les réseaux neuronaux ont vu le jour qu'en 1943 par W.MCCulloch et W. Pitts du neurone formel qui est une abstraction du neurone physiologique. Par cette

présentation, ils ont pu démontrer que le cerveau est équivalent à une machine de Turing, la pensée devient alors purement des mécanismes matériels et logiques. Ils déclarèrent en 1955 "Plus nous apprenons de choses au sujet des organismes, plus nous sommes amenés à conclure qu'ils ne sont pas simplement analogues aux machines, mais qu'en est-il de cette machine. La démonstration de McCulloch et Pitts a été l'un des acteurs importants de la création de la cybernétique.

En 1949, D. Hebb présenta dans son ouvrage "The Organization of Behavior" une règle d'apprentissage. De nombreux modèles de réseaux aujourd'hui s'inspirent encore de la règle de Hebb.

En 1958, F. Rosenblatt développe le modèle du Perceptron. C'est un réseau de neurones inspiré du système visuel. Il possède deux couches de neurones : une couche de perception et une couche liée à la prise de décision. C'est le premier système artificiel capable d'apprendre par expérience. Dans la même période, le modèle de L'Adaline (ADaptive LINar Element) a été présenté par B. Widrow, chercheur américain à Stanford. Ce modèle sera par la suite le modèle de base des réseaux multicouches.

En 1969, M. Minsky et S. Papert publient une critique des propriétés du Perceptron. Cela va avoir une grande incidence sur la recherche dans ce domaine. Elle va fortement diminuer jusqu'en 1972, où T. Kohonen présente ses travaux sur les mémoires associatives et propose des applications à la reconnaissance de formes.

C'est en 1982 que J. Hopfield présente son étude d'un réseau complètement rebouclé, dont il analyse la dynamique

Aujourd'hui, les réseaux neuronaux sont utilisés dans de nombreux domaines (entre autres, vie artificielle et intelligence artificielle) à cause de leur propriété en particulier, leur capacité d'apprentissage, et qu'ils soient des systèmes dynamiques [w15].

3.3 Topologie

Chaque réseau de neurones est connecté entre eux de diverses manières. De la figure suivante. Nous pouvons distinguer deux familles de réseaux de neurones : non bouclés ou statiques (a) et (b) et bouclés (dynamiques) (c) et (d).

4. L'apprentissage en profondeur (deep Learning)

Le Deep Learning est un nouveau domaine de recherche de la machine Learning (ML), qui a été introduit dans le but de rapprocher le ML de son objectif principal à savoir :

l'intelligence artificielle. Il concerne les algorithmes inspirés par la structure et du fonctionnement du cerveau. Ils peuvent apprendre plusieurs niveaux de représentation dans le but de modéliser des relations complexes entre les données.

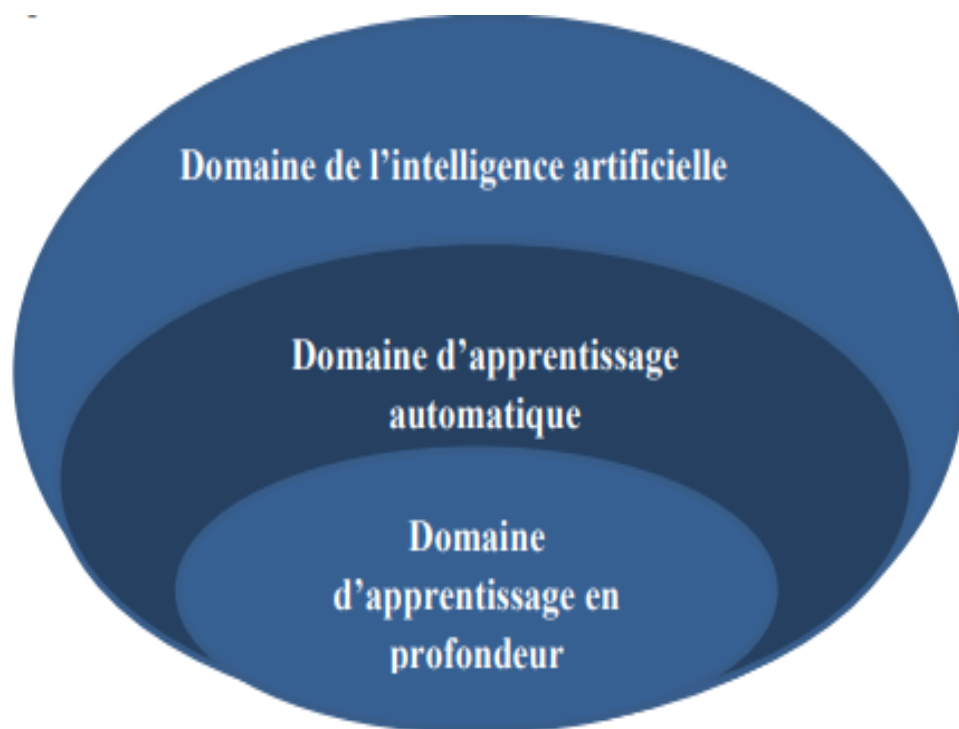


Figure 1. La relation entre l'intelligence artificielle, le ML et le deep Learning

4.1 Définition

L'apprentissage en profondeur est un ensemble d'algorithmes d'apprentissage automatique qui tentent d'apprendre à plusieurs niveaux, correspondant à différents niveaux d'abstraction. Il a la capacité d'extraire des caractéristiques à partir des données brutes grâce aux multiples couches de traitement composé de multiples transformations linéaires et non linéaires et apprendre sur ces caractéristiques petites à petit à travers chaque couche avec une intervention humaine minimale [14] [15] [16] [17] [18] [19] [20].

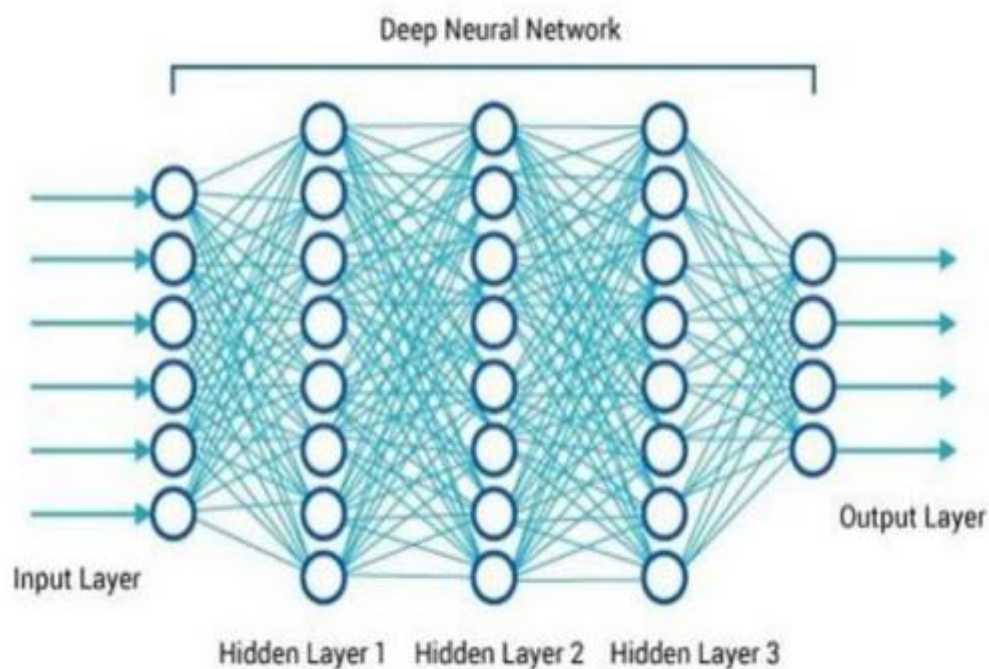


Figure 4. Schéma illustratif de DL avec plusieurs couches.

4.2 Historique

L'idée du Deep Learning n'est pas une idée récente, mais elle date en réalité des années 1980, plus particulièrement suite aux travaux de réseaux de neurones

multicouches et aux travaux de certains pionniers du machine Learning et du Deep Learning comme le français Yann Le Cun. En collaboration avec deux autres informaticiens, Kunihiko Fukushima et Geoffrey Hinton, ils mettent au point un type d'algorithme particulier appelé Convolutional neural network. Bien que cette approche donne des résultats, ses progrès et son évolution sont limités par les progrès technologiques en matière de micro-processeurs, de puissance de calculs, et du manque d'accessibilités à des données afin de pouvoir entraîner les neurones. Cependant certains chercheurs ont continué à travailler sur ce modèle pendant environ deux décennies et, avec l'aide des évolutions en matière de technologies, mais surtout avec la disponibilité toujours plus grande de données, ont pu améliorer cette technique.

Afin de développer un système d'apprentissage performant, il faut pouvoir l'exercer et cela requiert un nombre important de données à tester. C'est dans ce contexte qu'en 2007 le STANFORD VISION LAB, avec Fei-Fei Li à sa tête, développent un agrégateur d'images où sont consignés et étiquetés quelques millions de photos : ImageNet. En 2010, ImageNet regroupe 15 000 000 d'images toutes catégorisées en fonction de leurs caractéristiques propres (véhicules, animaux, ...) [w16].

En 2012, le Deep Learning est remis au goût du jour avec un succès retentissant au ImageNet Large Scale Visual Recognition Challenge (ILSVRC) qui est un concours annuel de reconnaissance d'image fondée par l'université de Stanford, dans le cadre de son laboratoire STANFORD VISION LAB [w16]. Plusieurs équipes de chercheurs en informatique s'affrontent dans ce concours tous les ans afin de décerner la victoire au programme ayant eu le plus faible taux d'échec. Et alors que les algorithmes d'apprentissage profond sont absents de la compétition, en 2012 c'est bel et bien un algorithme de Deep Learning qui va remporter l'édition 2012 à la surprise générale [w17].

4.3 Pour quoi le choix deep Learning

Tout d'abord les différents algorithmes du deep Learning ne sont apparus qu'à l'échec de l'apprentissage automatique tentant de résoudre une grande variété de problèmes de l'intelligence artificielle (l'IA) :

- Afin d'améliorer le développement des algorithmes traditionnels dans de telles tâches de l'IA.
- De développer une grande quantité de données telle que les big data.
- De s'adapter à n'importe quel type de problème.
- D'extraire les caractéristiques de façon automatique [21].

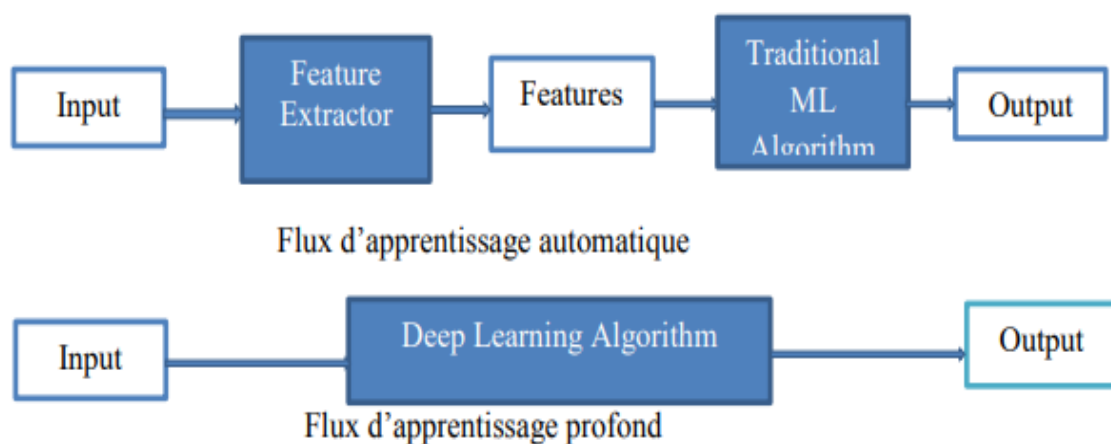


Figure 5. Comparaison entre la machine Learning et le Deep Learning

4.4 Les différentes Architectures du Deep Learning

Bien qu'il existe un grand nombre de variantes d'architectures profondes. Il n'est pas toujours possible de comparer les performances de toutes les architectures, car elles ne sont pas toutes évaluées sur les mêmes ensembles de données. Le Deep Learning est un domaine à croissance rapide, et de nouvelles architectures, variantes ou algorithmes apparaissent toutes les semaines

4.4.1 Les réseaux de neurones convolutifs

Convolutional Neural Network (CNN) (réseaux de neurones convolutifs) sont un type de réseau de neurones spécialisés pour le traitement de données ayant une topologie semblable à une grille. Qui se sont avérés très efficaces dans des domaines tels que la reconnaissance et la classification d'images et vidéos. CNN a réussi à identifier les visages, les objets, panneaux de circulation et auto-conduite des voitures [22]. Récemment, les CNN ont été efficaces dans plusieurs tâches de traitement du langage naturel (telles que la classification des phrases) [23]. [24] [25].

Dans le ML, un réseau convolutif est un type de réseau de neurones feed-forward, il a été inspiré par des processus biologiques [26]. Il existe quatre (4) principales opérations illustrées dans le CNN à savoir :

- La couche convolution.
- La couche Rectified Linear Unit.
- La couche Pooling.
- La couche entièrement connectée.

4.4.2 Réseau de neurones récurrents

L'idée derrière les RNN est d'utiliser des informations séquentielles. Dans un réseau neuronal traditionnel, nous supposons que toutes les entrées (et les sorties) sont indépendantes les unes des autres. Mais pour de nombreuses tâches, c'est une très mauvaise idée. Si on veut prédire le prochain mot dans une phrase, il faut connaître les mots qui sont venus avant. Les RNN sont appelés récurrents, car ils exécutent la même tâche pour chaque élément d'une séquence, la sortie étant dépendante des calculs précédents.

Une autre façon de penser les RNN est qu'ils ont une « mémoire » qui capture l'information sur ce qui a été calculé jusqu'ici. En théorie, les RNN peuvent utiliser des informations dans des séquences arbitrairement longues, mais dans la pratique, on les limite à regarder seulement quelques étapes en arrière [\[27\]](#) [\[28\]](#) [\[29\]](#) [\[30\]](#).

Il est utilisé pour :

- La modélisation du langage et génération de texte.
- La traduction automatique.
- La reconnaissance vocale.
- Et la description des images.

4.4.3 Modèle génératif

Si les modèles discriminatifs comme (CNN, RNN) sont utilisés pour prédire les données du label et de l'entrée, tant dis que le modèle génératif décrit comment générer les données, il apprend et fait des prédictions en utilisant la loi de Bayes [\[31\]](#). Cependant les modèles génératifs sont capables de bien plus que la simple classification comme par exemple générer de nouvelles observations. Voici quelques exemples de modèle génératif :

- Boltzmann Machines [\[32\]](#).
- Restricted Boltzmann Machines [\[33\]](#)[\[34\]](#).
- Deep Belief Networks [\[35\]](#).
- Deep Boltzmann Machines [\[36\]](#) [\[37\]](#).
- Generative Adversarial Networks [\[38\]](#).
- Generative Stochastic Networks [\[38\]](#).
- Adversarial autoencoders [\[39\]](#).

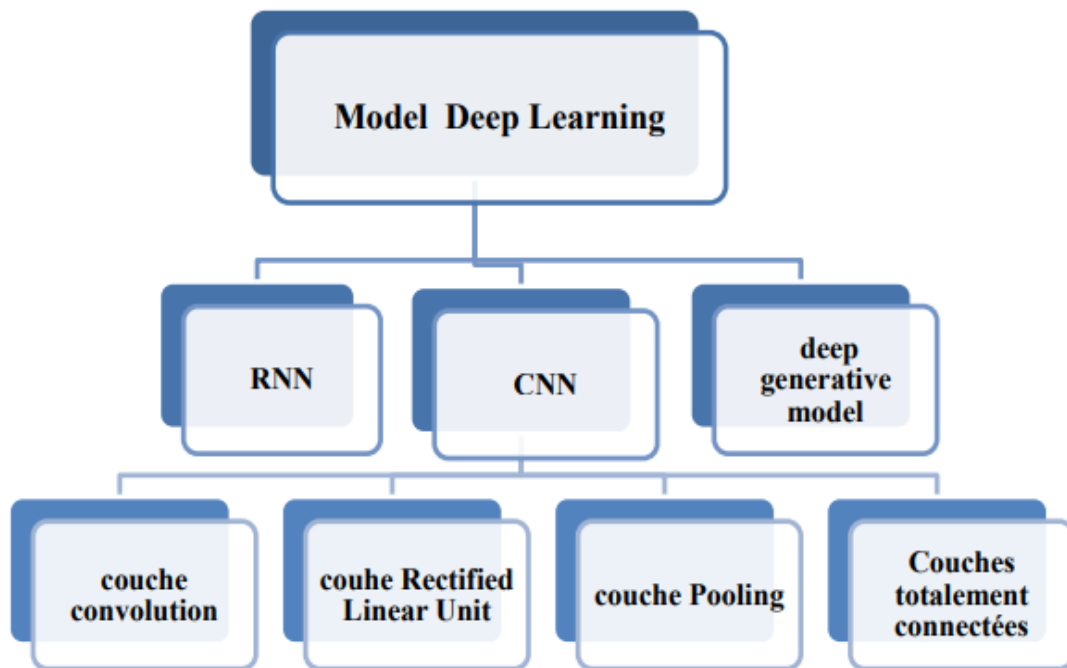


Figure 6. Différents modèles du Deep Learning.

4.4.4 Exemples d'application de Deep Learning

Les applications du Deep Learning sont utilisées dans divers secteurs, de la conduite automatisée aux dispositifs médicaux [w14].

Grace au deep Learning nous pouvons maintenant :

- Faire une colorisation des images en noir et blanc [40].
- Ajouter des sons à des films silencieux [41].
- Faire de la traduction automatique [42].
- Faire de classification des objets en photographies [43].
- Générer d'écriture automatique [44].
- Génération de légende d'image [45].

- Jeu automatique [46]

5. Convolutional Neural Networks, CNN

Les réseaux de neurones à convolution (Convolutional Neural Networks, CNN) sont une catégorie de réseaux de neurones qui se sont avérés très efficaces dans des domaines tels que la reconnaissance et la classification d'images. CNN ont réussi à identifier les visages, les objets, panneaux de circulation et auto-conduite des voitures c'est raison pour laquelle que durant cette phase de formation de nous nous baserons uniquement sur le CNN.

5.1 Architecture du système

Nous allons commencer par entraîner un réseau de neurones convolutif des plus classiques, qui possède en entrée une base de données et paramètres entraînables et il consiste en des couches convolutives, dont à chaque couche convolutives est placée une couche de Max pooling et enfin une couche totalement connectée. Pour un apprentissage plus rapide, ReLu sera la fonction d'activation.



Figure 2.1 : architecture générale d'un CNN

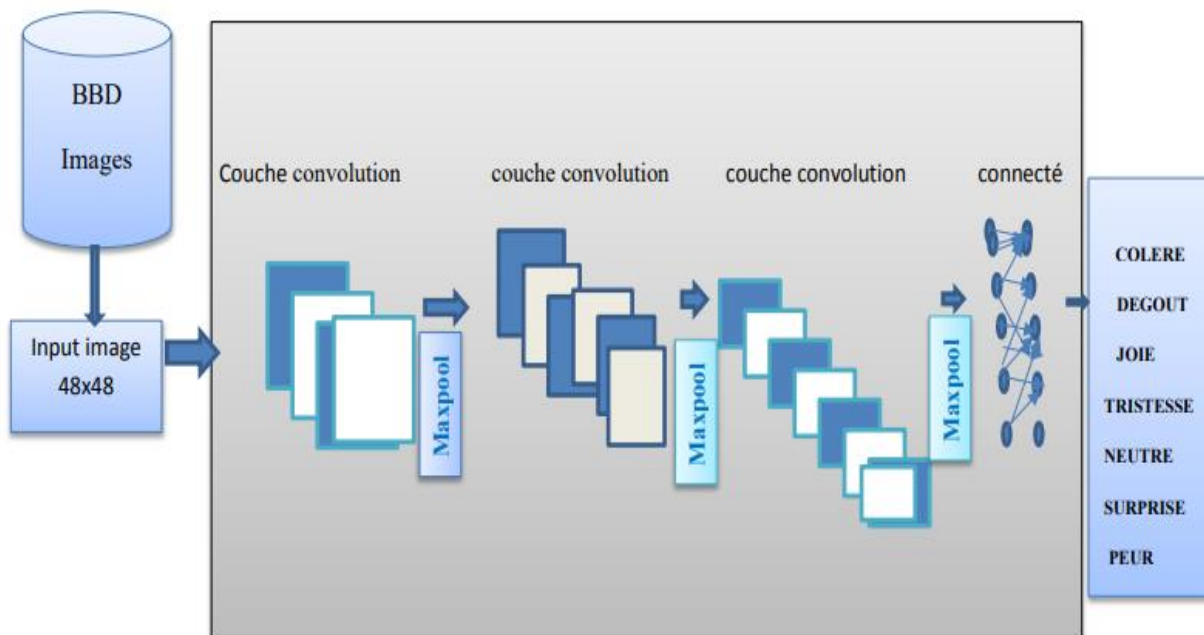


Figure 8. Architecture proposée.

5.2 Les différentes couches de CNN

Il existe quatre principales opérations dans le CNN, tout en ayant un rôle bien défini.

5.2.1 L'étape de convolution :

La convolution est le cœur du réseau de neurones convolutif, comme vous vous en doutez. À l'origine, une convolution est un outil mathématique (on parle de produit de convolution) très utilisé en retouche d'image, car il permet d'en faire ressortir l'extraction des caractéristiques à partir des images d'entrées, afin d'appliquer un bon filtre. En fait, une convolution prend simplement en entrée une image et un filtre (qui est une autre image), effectue un calcul, puis renvoie une nouvelle image (généralement plus petite) [47].

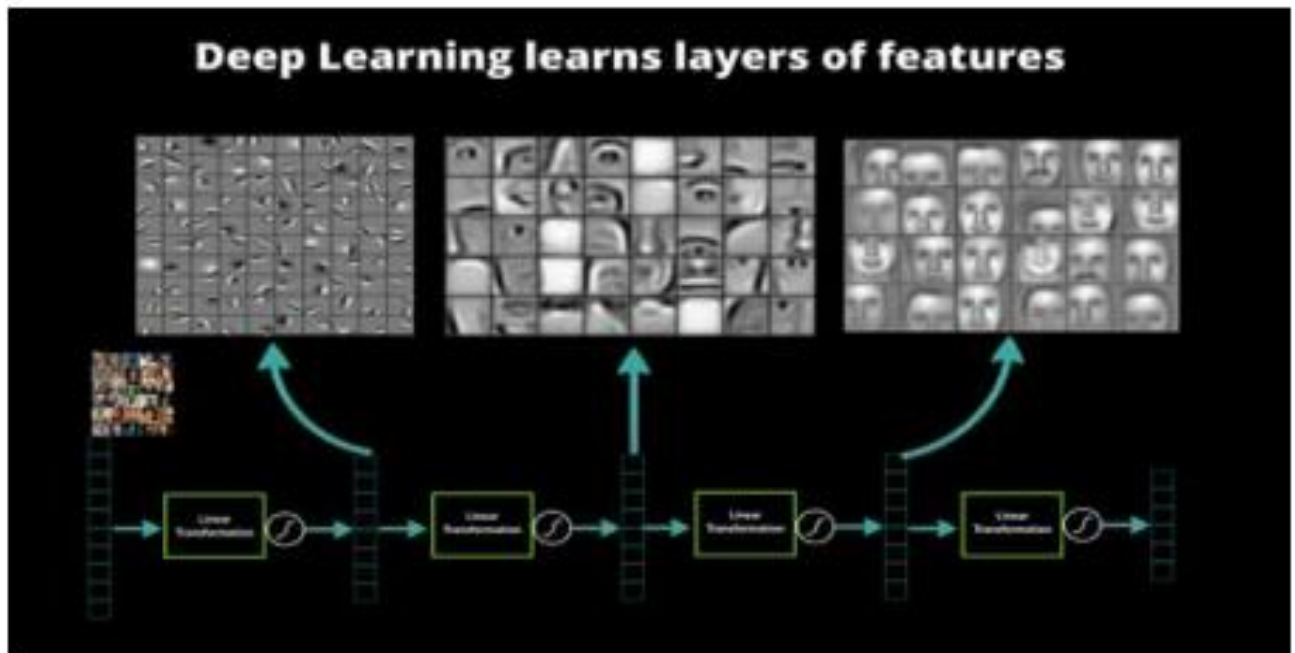


Figure9.Phase de convolution.

5.2.1.1 Les différentes convolutions

Il existe plusieurs types de convolutions, même si en général on utilise celle de base, il peut s'avérer utile de connaître les outils à notre disposition.

→ Convolution classique :

Qui représente le décalage du noyau entre chaque calcul, et le padding qui est la manière dont on peut « dépasser » de l'image pour appliquer la convolution.

→ Convolution dilatée :

Identique à la convolution à ceci près que le kernel est éclaté (on prend, par exemple, un pixel sur deux pour calculer la convolution). Il y a un paramètre supplémentaire : la dilation rate, qui est le nombre de pixels à ignorer.

→ Convolution transpose :

Qui construit la sortie comme si on inversait une convolution sur l'image

→ Convolution séparable :

Qui est une convolution décomposable en convolutions plus simples

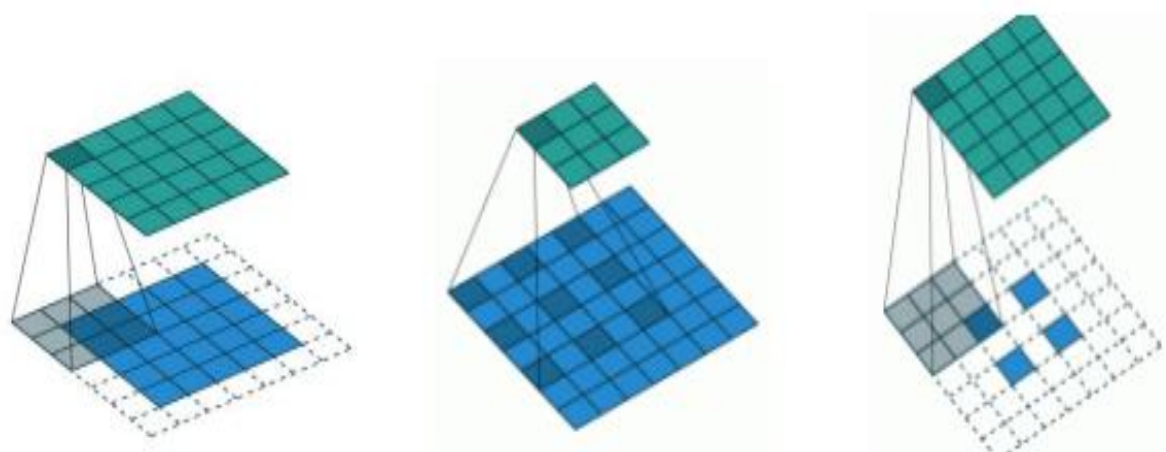


Figure2.4 : différents types de convolutions.

5.2.1.2 Fonctions d'activation

Une fonction d'activation est une fonction mathématique appliquée à un signal en sortie d'un neurone artificiel. Il dérive de l'équivalent biologique qui signifie "potentiel d'activation, lorsque le seuil de stimulation aura été atteint entraîne une réponse du neurone. Son but principal est de pouvoir permettre aux réseaux de neurones d'apprendre des fonctions plus complexes qu'une simple régression linéaire, car le simple fait de multiplier les poids d'une couche cachée est juste une transformation linéaire :

—Rectified Linear Unit (ReLU):

Elle est utilisée après chaque opération de convolution, ou toutes les valeurs de pixels négatifs sont mises à zéro. Le but de ReLU est d'introduire la non-linéarité dans notre CNN, puisque la plupart des données du monde réel, puisque la plupart de caractéristiques appliquées à l'une des cartes d'entrée donne une carte de sortie qui est également appelée carte de caractéristiques rectifiées [\[48\]](#).

5.2.2 L'étape de pooling

Elle permet de réduire la dimension de chaque carte de caractéristiques, mais conserve l'information la plus importante. IL peut être de différents types : max, moyenne, somme, etc. En cas de Pooling max, on définit un voisinage spatial (par exemple, une fenêtre 2x2) et de prendre le plus grand élément dans cette fenêtre.

La fonction de Pooling consiste à réduire progressivement la taille de la carte de caractéristiques rectifiée, en particulier, pooling :

- Rend les représentations d'entrée plus petites et plus faciles à gérer,
- Réduit le nombre de paramètres et les calculs dans le réseau [49].
- rend le réseau invariant aux petites transformations, les distorsions et les translations dans l'image d'entrée (une faible distorsion en entrée ne change pas la sortie de la Pooling car nous prenons le maximum / valeur moyenne dans un voisinage local).

Nous aide à arriver à une représentation presque invariante à l'échelle de notre image [50] [51].

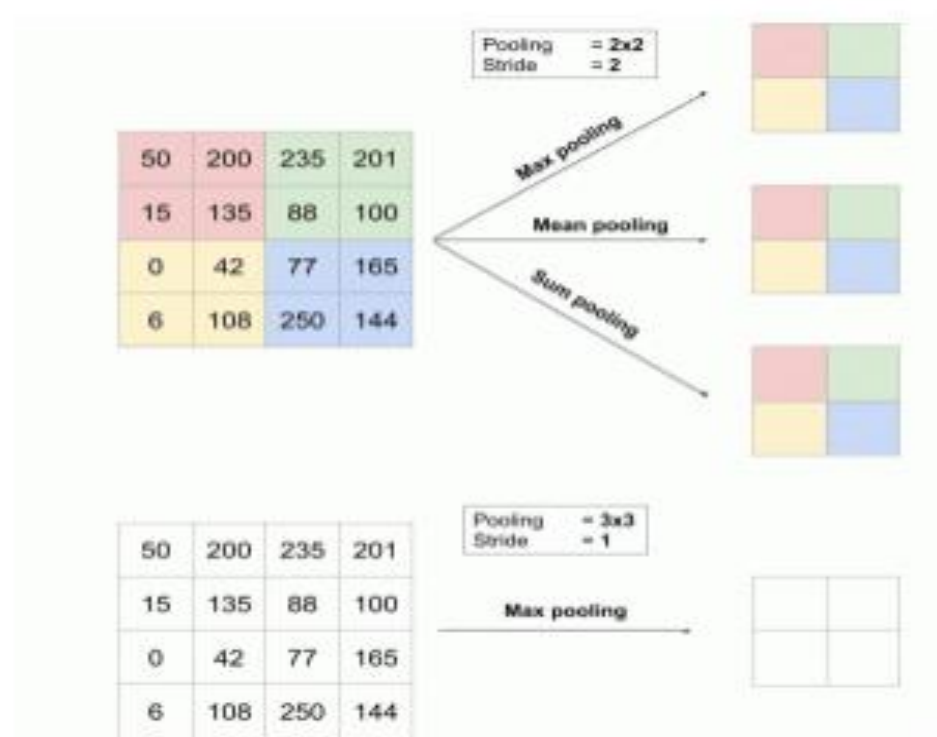


Figure 10. Exemple de description de la couche pooling.

Il existe plusieurs types de pooling :

- Le « **max pooling** », qui revient à prendre la valeur maximale de la sélection. C'est le type le plus utilisé, car il est rapide à calculer (immédiat), et permet de simplifier efficacement l'image
- Le « **mean pooling** » (ou **average pooling**), soit la moyenne des pixels de la sélection : on calcule la somme de toutes les valeurs et on divise par le nombre de valeurs. On obtient ainsi une valeur intermédiaire pour représenter ce lot de pixels
- Le « **sum pooling** », c'est la moyenne sans avoir divisé par le nombre de valeurs (on ne calcule que leur somme)

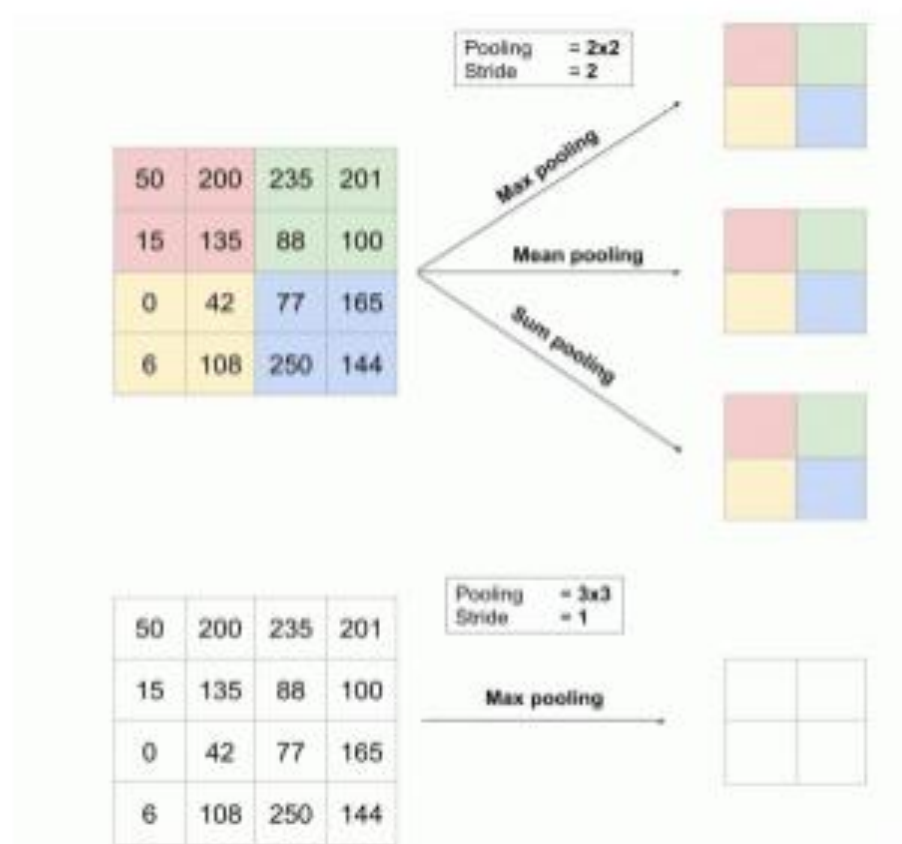


Figure 11. Calcul du pooling sur une image 4x4.

5.2.2.1 La mise à plat

Il consiste simplement à mettre à bout toutes les images (matrices) que nous avons pour en faire un (long) vecteur. Les pixels (en réalité ce ne sont plus des images ou des pixels, mais des matrices de nombres, donc les pixels sont ces nombres) sont récupérés ligne par ligne et ajoutés au vecteur final.

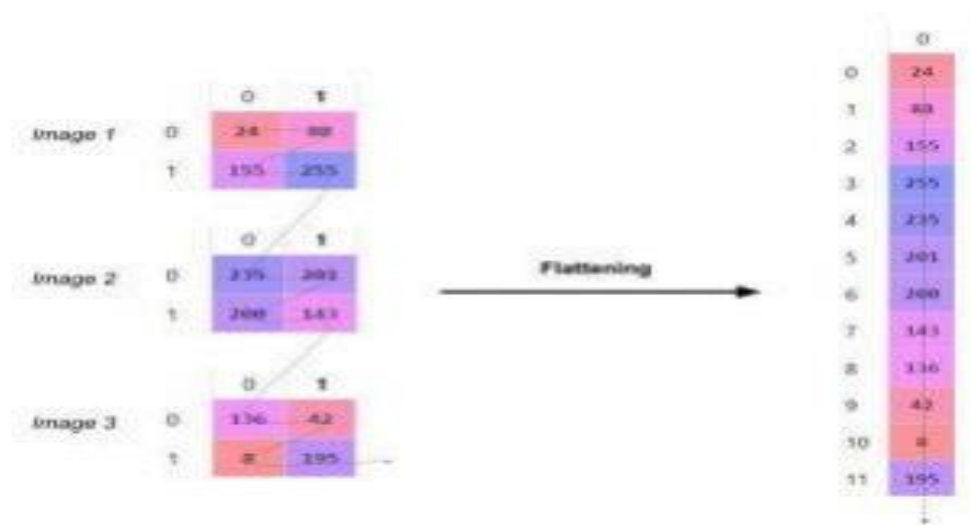


Figure 12. Mise à plat des images.

5.2.3 La couche totalement connectée

La couche entièrement connectée est un traditionnel perceptron multicouche (Multi Layer Perceptron) utilisant une fonction d'activation notamment appelée « soft max » dans la couche de sortie (d'autres classificateurs comme SVM peuvent également être utilisés). Le terme « entièrement connecté » implique que chaque neurone dans la couche précédente est connecté à chaque neurone sur la couche suivante. La sortie des couches de convolution et de Pooling représente les fonctions de haut niveau de l'image d'entrée. Le but de la couche entièrement connectée est de pouvoir utiliser ces fonctions pour classer l'image d'entrée dans différentes classes en fonction de l'ensemble de données d'apprentissage [52] .

6. Optimisation pour l'apprentissage en Deep Learning

a. La descente de gradient

La descente de gradient est un algorithme d'optimisation souvent utilisé pour trouver les poids ou les coefficients des algorithmes d'apprentissage automatique, tels que les réseaux de neurones artificiels et la régression logistique. Cela fonctionne en permettant au modèle de faire des prédictions sur les données d'apprentissage et en utilisant l'erreur sur les prédictions pour mettre à jour le modèle de manière à réduire l'erreur.

Le but de l'algorithme est de trouver des paramètres de modèle (par exemple, des coefficients ou des poids) qui minimisent l'erreur du modèle sur le jeu de données d'apprentissage. Pour ce faire, il modifie le modèle en le déplaçant le long d'une pente d'erreur vers une valeur d'erreur minimale. Cela donne à l'algorithme le nom de « descente de gradient » [53].

Il existe trois variantes de cette méthode :

- Batch gradient descent.
- Descente de gradient stochastique .
- Mini-batch gradient descent [w18].

6.1 Algorithmes d'optimisation de la descente de gradient

a. Adagrad : [54] est un algorithme d'optimisation basé sur la descente de gradient qui ne fait qu'adapter le taux d'apprentissage aux paramètres, effectuant de plus grandes mises à jour pour les caractéristiques peu fréquentes et de plus petites pour les caractéristiques plus fréquentes. Dean et al. [55] ont trouvé que Adagrad a grandement amélioré la robustesse de SGD et utilisé pour l'apprentissage de grands réseaux de neurones chez Google qui

-entre autres- ont appris à reconnaître les chats dans les vidéos YouTube [56]. De plus Pennington et al [57]. ont utilisé Adagrad pour l'apprentissage de GloVe Word embeddings, comme les mots les moins fréquents demandent de plus grandes mises à jour que ceux qui sont plus fréquents.

b. RMSprop : RMSprop est une méthode de taux d'apprentissage adaptative qui a été proposée par Geoff Hinton [58]. Au lieu d'accumuler tous les carrés des gradients précédents, nous allons tout simplement restreindre la fenêtre des gradients accumulés à une taille fixée w . Au lieu de stocker les w carrés des gradients précédents, on applique une moyenne mobile exponentielle des carrés des gradients précédents.

7. Conclusion

Dans ce chapitre. Nous avons eu à définir Qu'est-ce que le Deep Learning et comment il se différencie des algorithmes de ML traditionnelles.

Nous avons également parlé des trois familles majeures de modèle à savoir les réseaux convolutifs, les réseaux récurrents ainsi que les modèles génératifs, ces derniers font encore l'objet de recherches intensives. Après, nous rentrerons plus en détail sur les réseaux de neurones convolutifs (CNN).

Nous avons tout d'abord défini l'architecture utilisée et aussi essayé d'expliquer les concepts principaux de réseau de neurone à convolution (CNN) en termes simples. Le CNN à quatre principales opérations : convolution, la fonction non-linéarité (ReLU), Pooling et couche entièrement connectée. Première opération est la convolution pour l'extraction de caractéristiques de l'image d'entrée. Deuxième opération est la fonction non linéarité (ReLU) pour remplacer toutes les valeurs de pixels négatives par zéro. Troisième opération est la Pooling pour réduire progressivement la taille de la carte de caractéristiques rectifiée. Enfin une couche entièrement connectée pour la classification. Et aussi nous vous présentons quelques méthodes permettant la réduction des erreurs

Chapitre III

L'implémentation

1. Introduction

L'objectif de ce chapitre est de présenter les étapes de l'implémentation de l'approche proposée dans le cadre d'un système de reconnaissance faciale de visage. Nous commençons tout d'abord par la présentation des ressources, du langage et de l'environnement de développement que nous avons utilisé. Puis les étapes de la réalisation du modèle et on termine par les tests effectués. Ce chapitre est composé de deux parties, l'implémentation du système et les résultats expérimentaux des tests.

2. Environnement de développement

2.1 Visual Studio Code

Est une version réduite de l'environnement officiel de développement Microsoft qui se concentre exclusivement sur l'éditeur de **code**. Il est multiplateforme et supporte des syntaxes pour un grand nombre de langages.

2.2 Comment mettre son Visual Studio Code ?

Appuyez sur Ctrl+Maj+P pour faire apparaître la Palette de commandes, puis commencez à taper afficher pour filtrer le contenu et afficher la commande Configurer la langue d'affichage. Appuyez sur Entrée pour afficher une liste de langues installées en fonction des paramètres régionaux.

2.3 Pourquoi utiliser Visual Studio Code ?

L'un des outils le plus important de **VS Code** est la capacité de déboguer les applications directement sur l'éditeur sans même avoir recours aux navigateurs, grâce à un système de points d'arrêt et une console de débogage intégrée qui permet de résoudre les problèmes directement dans l'éditeur.

2.4 Python :

Python est un langage de programmation de haut niveau utilisé pour la programmation générale. Créé par Guido van Rossum et sorti en 1991, Python a une philosophie de conception qui met l'accent sur la lisibilité du code, notamment en utilisant des espaces importants. Il fournit des constructions qui permettent une programmation claire à petite et à grande échelle. Python dispose d'un système de type dynamique et d'une gestion automatique de la mémoire. Il prend en charge de multiples paradigmes de programmation, y compris orientés objet, impératifs, fonctionnels et procéduraux, et dispose d'une bibliothèque standard vaste et complète. Les interpréteurs de Python sont disponibles pour de nombreux systèmes d'exploitation. [W7]



Figure 3.1 Le logo de python.

2.5 À quoi sert Visual Studio Code ?

Visual Studio Code est un éditeur de code simplifié qui prend en charge les opérations de développement comme le débogage, l'exécution de tâches et le contrôle de version. Il vise à fournir juste les outils dont un développeur a besoin pour un cycle rapide de code-build-debugge et laisse des workflows plus complexes à des IDEs plus complets, tels que Visual Studio IDE.

3. Virtual Environnement :

Un environnement virtuel est un environnement Python tel que l'interpréteur Python, les bibliothèques et les scripts installés sont isolés de ceux installés dans d'autres environnements virtuels, et (par défaut) de toutes autres bibliothèques installées dans un Python "système", par exemple celui qui est installé avec votre système d'exploitation.

Un environnement virtuel est une arborescence de dossiers qui contiens les fichiers exécutables Python et autres fichiers qui indiquent que c'est un environnement virtuel.

Les outils d'installations communs comme [setup Tools](#) et [pip](#) fonctionnent comme prévu avec des environnements virtuels. En d'autres termes, quand un environnement virtuel est actif, ils installent les paquets Python dans l'environnement virtuel sans avoir besoin de leur préciser explicitement.

Quand un environnement virtuel est actif (Par exemple quand l'interpréteur Python de l'environnement virtuel est lancé), les attributs `sys.prefix` et `sys.exec_prefix` pointent vers le dossier racine de l'environnement virtuel, alors que `sys.base_prefix` et `sys.base_exec_prefix` pointent vers l'installation de Python qui n'est pas celle de l'environnement virtuel et qui a été utilisée pour créer l'environnement virtuel. Si un environnement virtuel n'est pas actif, alors `sys.prefix` est égal à `sys.base_prefix` et `sys.exec_prefix` est égal à `sys.base_exec_prefix` (ils pointent tous sur une installation Python qui n'est pas un environnement virtuel).

Quand un environnement virtuel est actif, toute option qui change le chemin d'installation sera ignorée de tous les fichiers de configuration [distutils](#) pour éviter que des projets soient accidentellement installés en dehors de l'environnement virtuel.

Quand vous travaillez dans une invite de commande **Shell**, les utilisateurs peuvent activer un environnement virtuel en lançant un script `activate` situé dans le dossier des exécutables de l'environnement virtuel (le nom de fichier précis dépend du Shell utilisé), ce qui ajoute le dossier des exécutables de l'environnement virtuel dans la variable d'environnement `PATH`. Il ne devrait jamais y avoir besoin dans d'autre circonstances d'activer un environnement virtuel.

Des scripts installés dans un environnement virtuel ont un « *shebang* » qui pointe vers l'interpréteur Python de l'environnement virtuel. Cela veut dire que le script sera lancé avec cet interpréteur peut importer la valeur de `PATH`. Sur Windows, le « *shebang* » est interprété si vous avez le Lanceur Python pour Windows installé (Cela a été ajouté à Python 3.3 -- Voir [PEP 397](#) pour plus de détails). De ce fait, double cliquer un script installé dans une fenêtre de l'Explorateur Windows devrait lancer le script avec le bon interpréteur sans avoir besoin de référencer son environnement virtuel dans `PATH`.

3.1 Pourquoi utiliser environnement virtuel Python ?

Les **environnements virtuels Python** permettent d'avoir des installations de **Python** isolées du système et séparées les unes des autres. Cela permet de gérer plusieurs projets sur sa machine de développements, certains utilisant des modules de versions différentes, voir même des versions différentes de **Python**.

3.2 Comment activer son environnement virtuel Python ?

Exécutez l'application dans un **environnement virtuel**

Pour créer un **environnement**, utilisez la commande `python -m venv`

`<environnement Name>`. Si vous exécutez à nouveau `pip freeze`, vous ne devriez maintenant plus voir de paquet répertorié. Cela montre qu'aucun paquet n'est installé dans votre **environnement virtuel**.

3.3 Création d'environnements virtuels :

La création d'**environnements virtuels** est faite en exécutant la commande `venv` :

```
python3 -m venv /path/to/new/virtual/environment
```

Lancer cette commande crée le dossier cible (en créant tous les dossiers parents qui n'existent pas déjà) et y ajoute un fichier `pyvenv.cfg` contenant une variable `home` qui pointe sur l'installation Python depuis laquelle cette commande a été lancée (un nom habituel pour ce dossier cible est `.venv`). Cela crée également un sous-dossier `bin` (ou `Scripts` sous Windows) contenant une copie (ou un lien symbolique) du ou des binaires `python`s (dépend de la plateforme et des paramètres donnés à la création de l'environnement). Elle crée aussi un sous-dossier (initialement vide) `lib/pythonX.Y/site-packages` (Sous Windows, c'est `Lib\site-packages`). Si un dossier existant est spécifié, il sera réutilisé.

Obsolète depuis la version 3.6: `pyvenv` était l'outil recommandé pour créer des environnements sous Python 3.3 et 3.4, et est [obsolète depuis Python 3.6](#).

Modifié dans la version 3.5: L'utilisation de `venv` est maintenant recommandée pour créer vos environnements virtuels.

Sur Windows, appelez la commande `venv` comme suit : `C:\Users\salam\python -m venv C:\Users\salam\desktop\Facerecognition\myvenvpy`

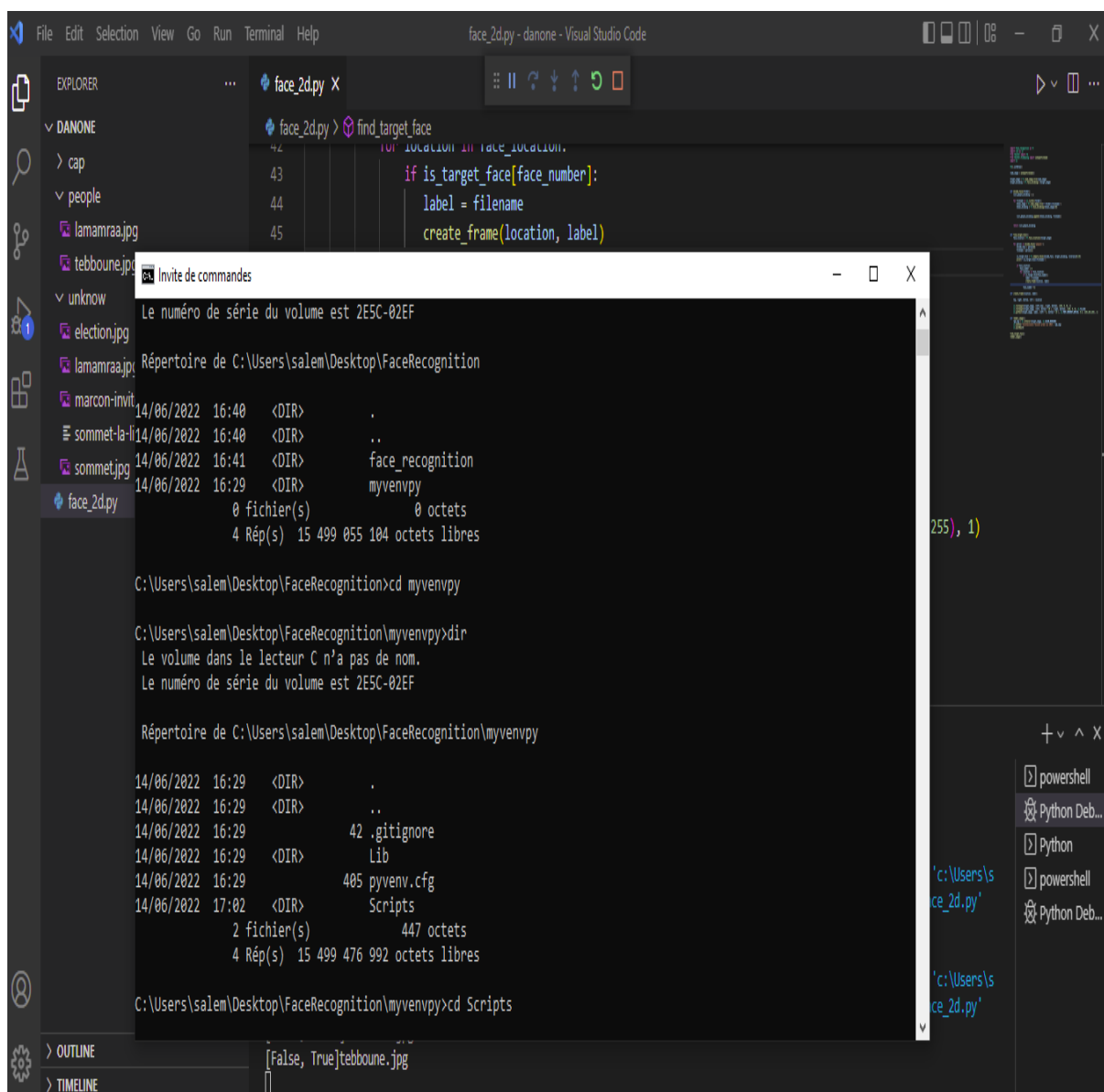


Figure 3.2 création d'environnements virtuels.

Alternativement, la configuration des deux variables `PATH` et `PATHEXT` pour l'installation Python :

```
c:\>python -m venv c:\path\to\myenv
```

La commande lancée avec `-h`, montrera les options disponibles :

```
usage: venv [-h] [--system-site-packages] [--symlinks | --copies] [--clear]
           [--upgrade] [--without-pip] [--prompt PROMPT] [--upgrade-deps]
           ENV_DIR [ENV_DIR ...]

Creates virtual Python environments in one or more target directories.

positional arguments:
  ENV_DIR                A directory to create the environment in.

optional arguments:
  -h, --help            show this help message and exit
  --system-site-packages
                        Give the virtual environment access to the system
                        site-packages dir.
  --symlinks            Try to use symlinks rather than copies, when symlinks
                        are not the default for the platform.
  --copies              Try to use copies rather than symlinks, even when
                        symlinks are the default for the platform.
  --clear               Delete the contents of the environment directory if it
                        already exists, before environment creation.
  --upgrade             Upgrade the environment directory to use this version
                        of Python, assuming Python has been upgraded in-place.
  --without-pip         Skips installing or upgrading pip in the virtual
                        environment (pip is bootstrapped by default)
  --prompt PROMPT       Provides an alternative prompt prefix for this
                        environment.
  --upgrade-deps        Upgrade core dependencies: pip setuptools to the
                        latest version in PyPI

Once an environment has been created, you may wish to activate it, e.g. by
sourcing an activate script in its bin directory.
```

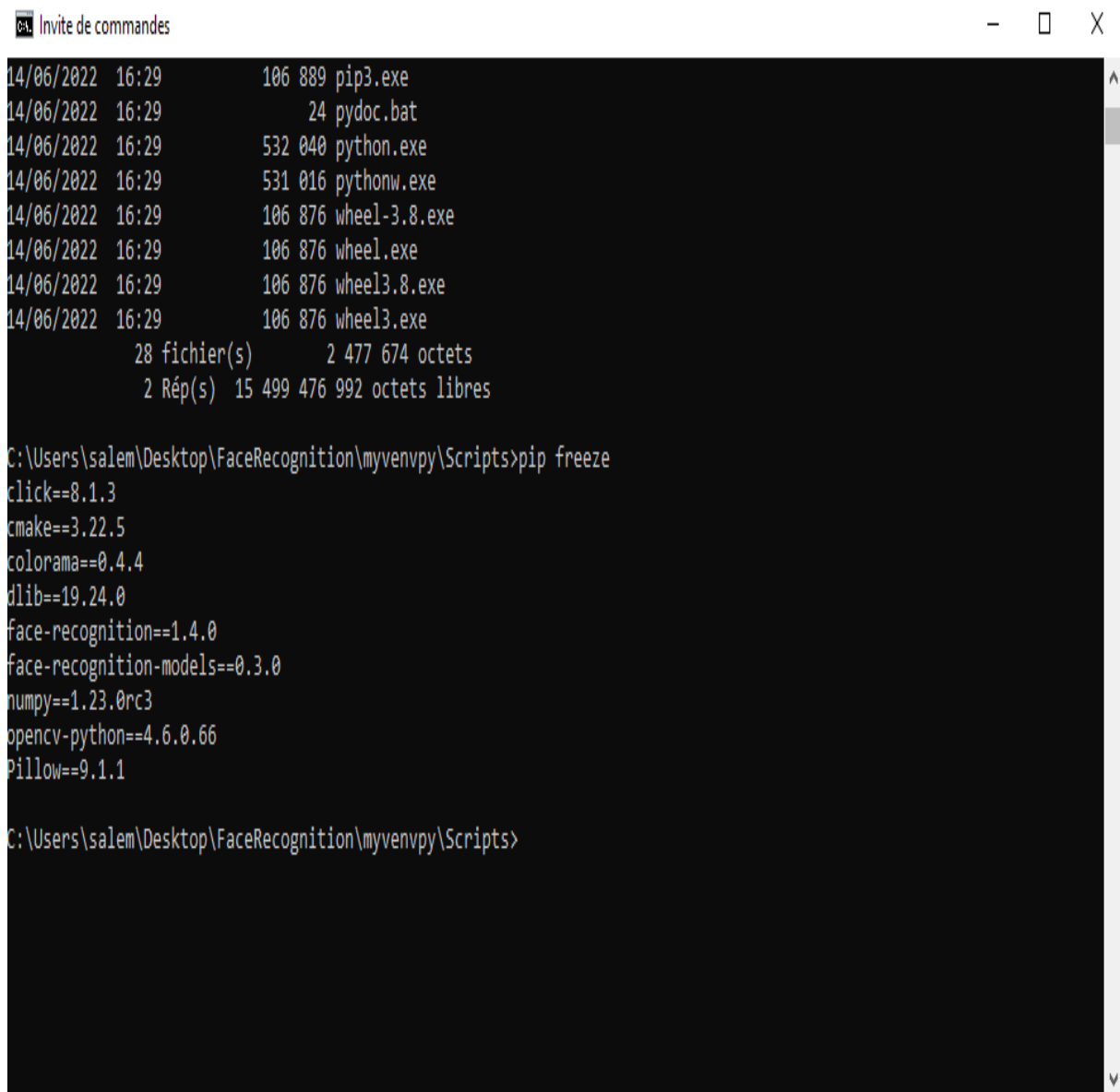
Figure 3.3 création environnement virtuel de python.

Plusieurs chemins peuvent être donnés à `venv`, et dans ce cas un environnement virtuel sera créé, en fonction des options incluses, à chaque chemin donné.

4. Bibliothèques utilisées :

Pour afficher l'ensemble des bibliothèques implémenter dans le code on exécute la commande

Pip List dans le terminal ou pip freezer comme suit :



```
14/06/2022 16:29      106 889 pip3.exe
14/06/2022 16:29         24 pydoc.bat
14/06/2022 16:29     532 040 python.exe
14/06/2022 16:29     531 016 pythonw.exe
14/06/2022 16:29     106 876 wheel-3.8.exe
14/06/2022 16:29     106 876 wheel.exe
14/06/2022 16:29     106 876 wheel3.8.exe
14/06/2022 16:29     106 876 wheel3.exe
      28 fichier(s)      2 477 674 octets
      2 Rép(s) 15 499 476 992 octets libres

C:\Users\salem\Desktop\FaceRecognition\myvenvpy\Scripts>pip freeze
click==8.1.3
cmake==3.22.5
colorama==0.4.4
dlib==19.24.0
face-recognition==1.4.0
face-recognition-models==0.3.0
numpy==1.23.0rc3
opencv-python==4.6.0.66
pillow==9.1.1

C:\Users\salem\Desktop\FaceRecognition\myvenvpy\Scripts>
```

Figureur 3.4 les bibliothèques.

4.1 Face recognition :

Reconnaître et manipuler les visages de Python ou de la ligne de commande avec la bibliothèque de reconnaissance faciale la plus simple au monde. Construit en utilisant la reconnaissance faciale de pointe de dlib Construit avec l'apprentissage profond. Le modèle a une précision de 99,38% sur le labeléd Faces in the Wild.

Cela fournit également un outil de ligne de commande face recognition simple qui permet

Vous faites face à la reconnaissance sur un dossier d'images de la ligne de commande!

- **Caractéristiques**

- **Trouver des visages en images**

Trouver tous les visages qui apparaissent dans une image :



Figure 3.5 Input.



Figure 3.6 Out put.

```
import face_recognition
image = face_recognition.load_image_file("your_file.jpg")
face_locations = face_recognition.face_locations(image)
```

➤ **Trouver et manipuler les traits du visage dans les images**

Obtenez les emplacements et les contours des yeux, du nez, de la bouche et du menton de chaque personne.

**Figure 3.7 Input.****Figure 3.8 output.**

```
import face_recognition
image = face_recognition.load_image_file("your_file.jpg")
face_landmarks_list = face_recognition.face_landmarks(image)
```

Trouver les traits du visage est super utile pour beaucoup de choses importantes. Mais vous pouvez aussi utiliser pour des trucs vraiment stupides Comme appliquer du maquillage numérique (pensez « Meitu ») :

Identifier les visages sur les photos

Reconnaissez qui apparaît sur chaque photo.

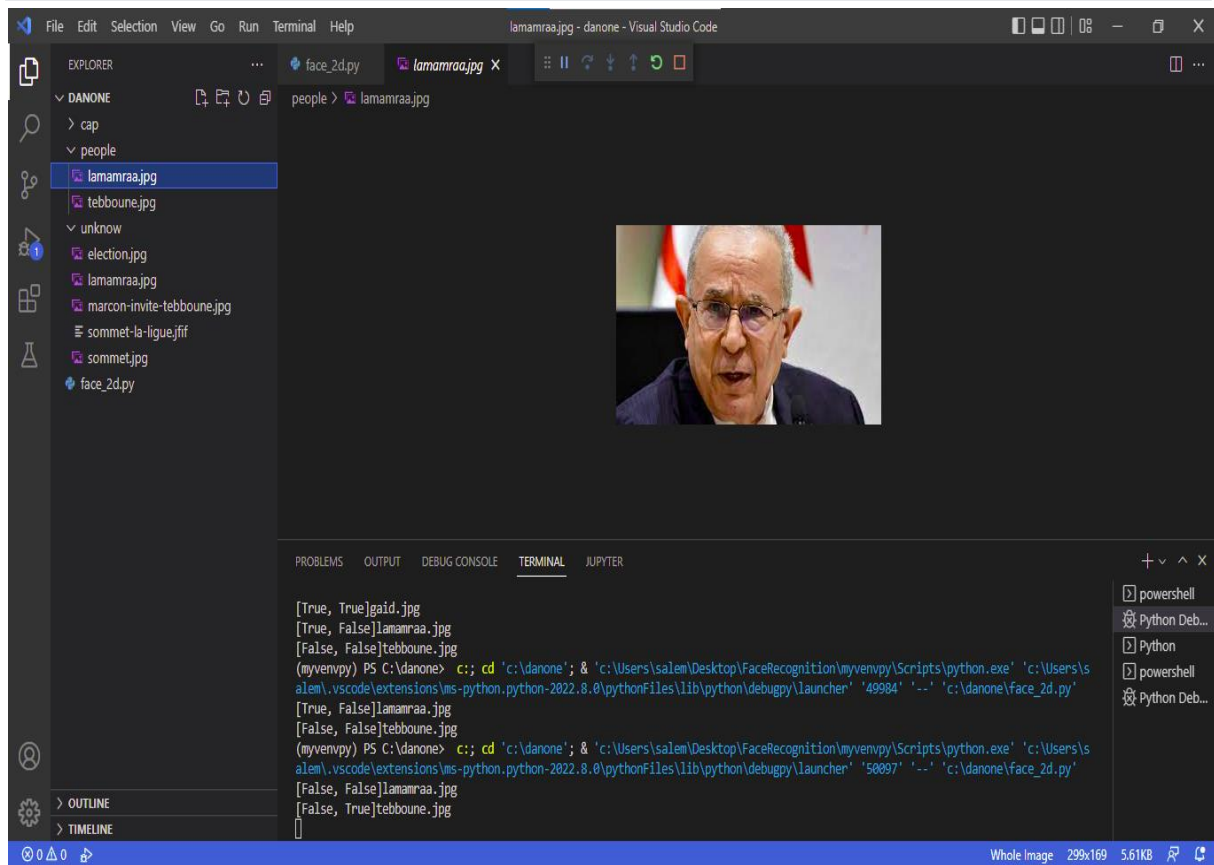


Figure 3.9 Input

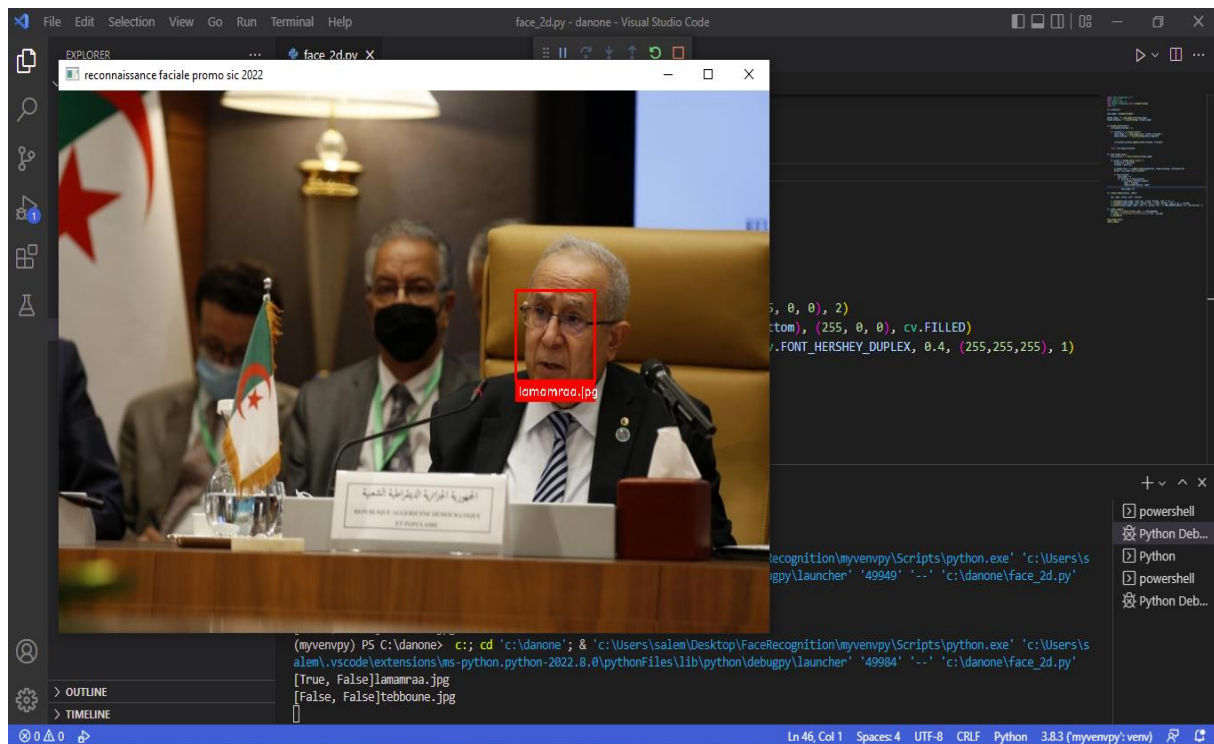


Figure 3.10 Output.

```
import face_recognition
known_image = face_recognition.load_image_file("biden.jpg")
unknown_image = face_recognition.load_image_file("unknown.jpg")

biden_encoding = face_recognition.face_encodings(known_image)[0]
unknown_encoding = face_recognition.face_encodings(unknown_image)[0]

results = face_recognition.compare_faces([biden_encoding], unknown_encoding)
```

4.2 Open CV (Open Source Computer Vision Library) :

Open CV (Open Source Computer Vision Library) est une bibliothèque de logiciels open source de vision par ordinateur et d'apprentissage automatique. qui est conçu pour fournir une infrastructure commune pour les algorithmes d'apprentissage automatique et la vision par ordinateur. Il dispose de milliers d'algorithmes optimisés qui peuvent être utilisés à différentes fins, comme la détection et la reconnaissance de visages, l'identification d'objets et bien d'autres. Nous en avons besoin pour prendre des photos à l'aide de notre webcam et quelques manipulations doivent être effectuées dans l'image.

Pour installer la bibliothèque, vous devez installer pip dans votre système, après quoi vous pouvez suivre les étapes de l'invite de commande :

Étape 1 : pip Install open cv-python

Étape 2 : pip Install open cv-contrib-python

4.3 NumPy :

NumPy est le package fondamental pour le calcul scientifique en Python qui fournit un objet de array multidimensionnel qui a été formé.

Pour installer la bibliothèque, vous pouvez taper une simple ligne de code dans votre Shell de commande :

```
pip install numpy
```

Nomme il existe plusieursbibliothèques fortement recommander pour la reconnaissance faciale quand n'a pas pu utiliser a défaut de ressources demander. Trop gourmand en processionnécessite le GPU et TPU Google pour un meilleurrésultat.

5. TensorFlow :

TensorFlow est un outil open source d'apprentissage automatique développé par Google. Le code source a été ouvert le 9 novembre 2015 par Google et publié sous licence Apache. Il est fondé sur l'infrastructure DistBelief, initiée par Google en 2011, et est doté d'une interface pour Python,

5.1 Comment fonctionne TensorFlow ?

Résultat de recherche d'images

L'interaction avec TensorFlow s'effectue par le biais du langage Python, à la fois simple à apprendre et à utiliser. Ce langage permet d'exprimer facilement comment les abstractions de haut-niveau peuvent être couplées ensemble. Les nœuds et les ténors de TensorFlow sont des objets Python.

5.2 Installation de TensorFlow :

Étape 1 : Vérifiez la version de python : `$ python3 --version`.

Étape 2 : Vérifiez si l'infusion est installée : ...

Étape 3 : Créez l'environnement virtuel : `$ brew install virtualenv`.

Étape 4 : Après avoir créé un nouvel environnement virtuel, créez un répertoire. `/pythonenv` pour le conserver.

6. Haar Cascade :

Haar Cascade est essentiellement un classificateur utilisé pour détecter les objets pour lesquels il a été formé, à partir de la source. Le résultat est un fichier XML qui stocke le résultat formé. Si cela est dit simplement, la cascade de Haar est formée en superposant l'image positive sur un ensemble d'images négatives. La formation nécessite un système de haute spécification et une bonne connexion Internet et des milliers d'images de formation, c'est pourquoi elle est réalisée sur le serveur. Pour augmenter l'efficacité des résultats, ils utilisent des images de haute qualité et augmentent le nombre d'étapes pour lesquelles le classificateur est entraîné. Nous avons besoin du reconnaisseur de visage frontal Haar Cascade pour détecter le visage de notre webcam.

7. Matplotlib :

Est une bibliothèque de traçage pour le langage de programmation Python et son extension mathématique numérique NumPy. Il fournit une API orientée objet permettant d'incorporer des graphiques dans des applications à l'aide de kits d'outils d'interface graphique à usage général tels que Tkinter, wxPython, Qt ou GTK +.

8. Keras Keras :

Est une bibliothèque open source écrite en python et permettant d'interagir avec les algorithmes de réseaux de neurones profonds et de machine **Learning**, notamment Tensorflow et Theano. Elle a été initialement écrite par François Chollet. [\[W11\]](#)

9. La phase apprentissage

Afin de faire la partie apprentissage nous avons commencé à entraîner notre algorithme à reconnaître au sein d'une base d'apprentissage des motifs récurrents. (Trois photo dans le répertoire people) Cet entraînement aboutit à un modèle informatique conçu pour faire des prédictions (reconnaître le visage de la personne en question...) . Par contre un programme traditionnel exécute des instructions, un algorithme de machine Learning améliore ses performances au fur et à mesure de son apprentissage, mais aussi au fil de l'évolution du contexte et de réentraînements successifs. Plus on le "nourrit" de données, plus il devient précis.

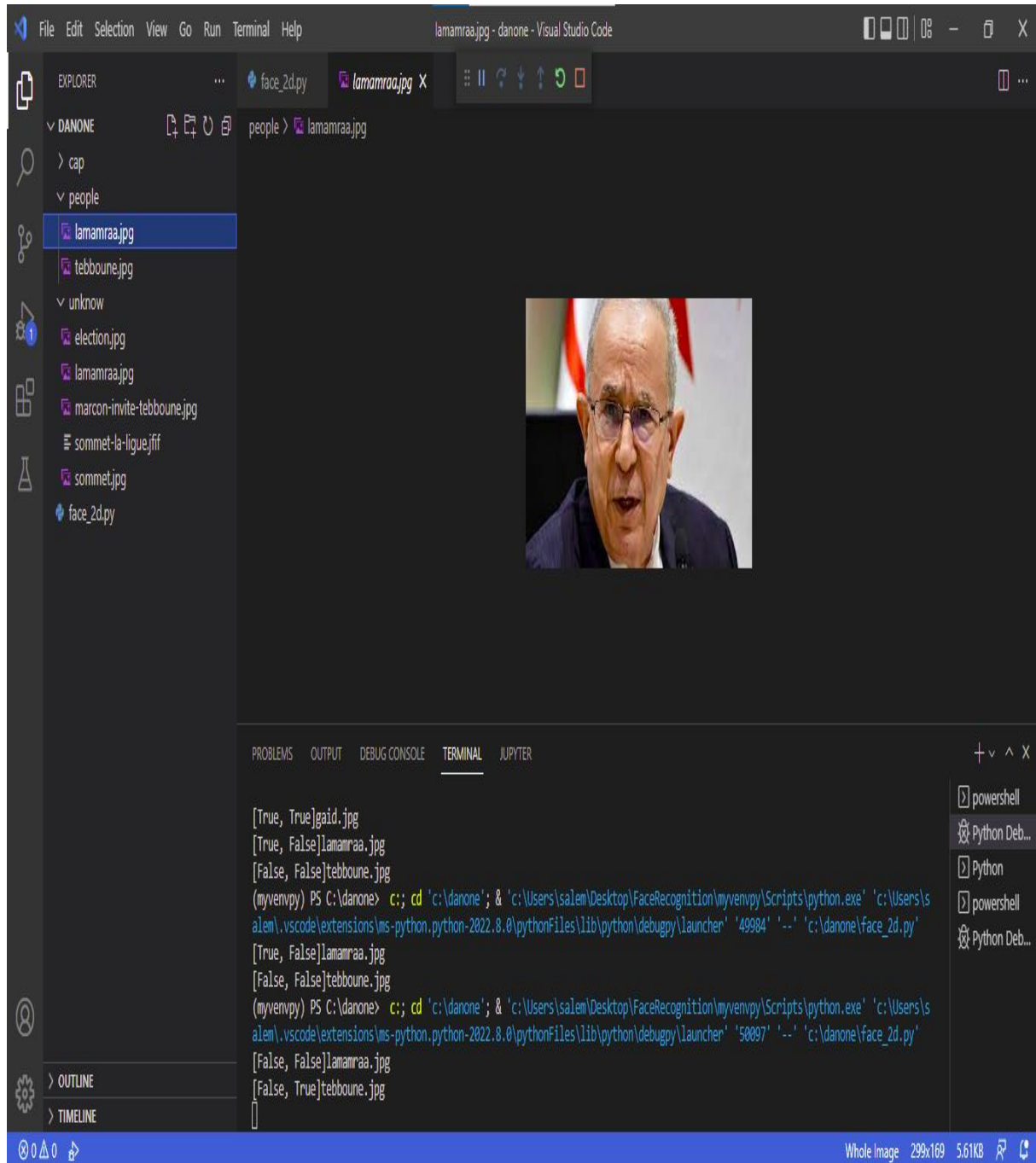


Figure 3.11 photo de lamamra.

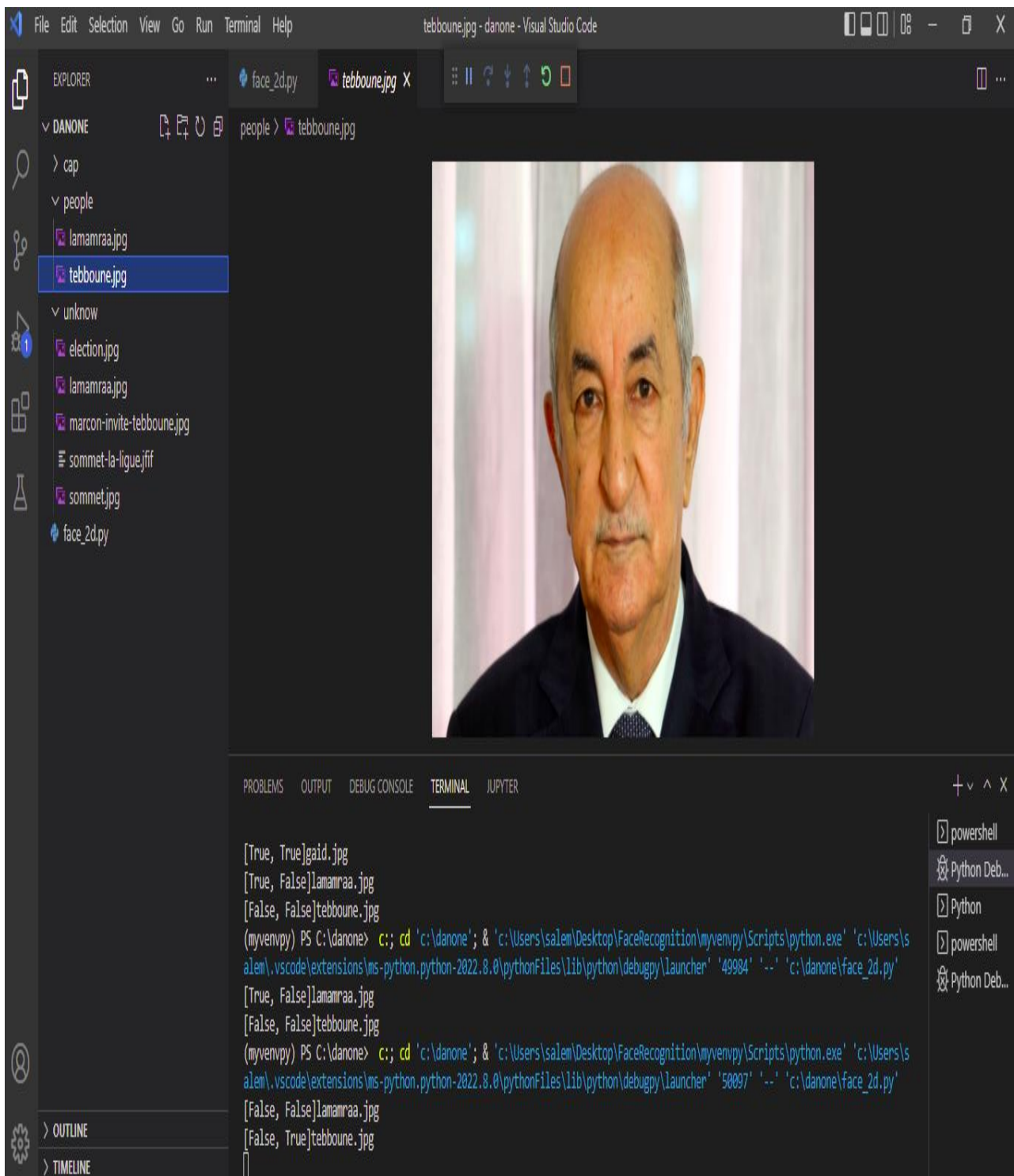


Figure 3.12 photo de tebboune.

10. La Phase de tests unitaire :

The screenshot shows the Visual Studio Code interface with the file `face_2d.py` open. The Explorer sidebar on the left shows a project named `DANONE` with subfolders `cap`, `people`, and `unknown`, each containing image files. The `face_2d.py` file is selected in the Explorer.

The code in `face_2d.py` is as follows:

```

42 for location in face_location:
43     if is_target_face[face_number]:
44         label = filename
45         create_frame(location, label)
46
47     face_number += 1
48
49 def create_frame(location, label):
50
51     top, right, bottom, left = location
52
53     cv.rectangle(target_image, (left,top), (right, bottom), (255, 0, 0), 2)
54     cv.rectangle(target_image, (left, bottom + 20 ), (right, bottom), (255, 0, 0), cv.FILLED)
55     cv.putText(target_image, label, (left + 3, bottom + 14 ), cv.FONT_HERSHEY_DUPLEX, 0.4, (255,255,255), 1)
56
57 def render_image():
58     rgb_img = cv.cvtColor(target_image, cv.COLOR_BGR2RGB)
59     cv.imshow('reconnaissance faciale promo sic 2022', rgb_img)
60     cv.waitKey(0)
61

```

The terminal at the bottom shows the execution of the unit tests:

```

[True, True]gaid.jpg
[True, False]lamamraa.jpg
[False, False]tebboune.jpg
(myvenvpy) PS C:\danone> c:: cd 'c:\danone'; & 'c:\Users\salem\Desktop\FaceRecognition\myvenvpy\Scripts\python.exe' 'c:\Users\salem\.vscode\extensions\ms-python.python-2022.8.0\pythonFiles\lib\python\debugpy\launcher' '49984' '-' 'c:\danone\face_2d.py'
[True, False]lamamraa.jpg
[False, False]tebboune.jpg
(myvenvpy) PS C:\danone> c:: cd 'c:\danone'; & 'c:\Users\salem\Desktop\FaceRecognition\myvenvpy\Scripts\python.exe' 'c:\Users\salem\.vscode\extensions\ms-python.python-2022.8.0\pythonFiles\lib\python\debugpy\launcher' '50897' '-' 'c:\danone\face_2d.py'
[False, False]lamamraa.jpg
[False, True]tebboune.jpg

```

The status bar at the bottom indicates the current position is `Ln 46, Col 1` with `Spaces: 4`, `UTF-8` encoding, `CRLF` line endings, and the Python version is `3.8.3 (myvenvpy:venv)`.

Figure 3.13 Capture de code de la phase de teste.

Après l'importation de différentes bibliothèque (os, face recogintion, open cv,cv2) et le mapping du l'environnement Virtual myvenvpy on a commencé à faire uploader les images un know pour faire tester notre algorithme de reconnaissance comme suit :

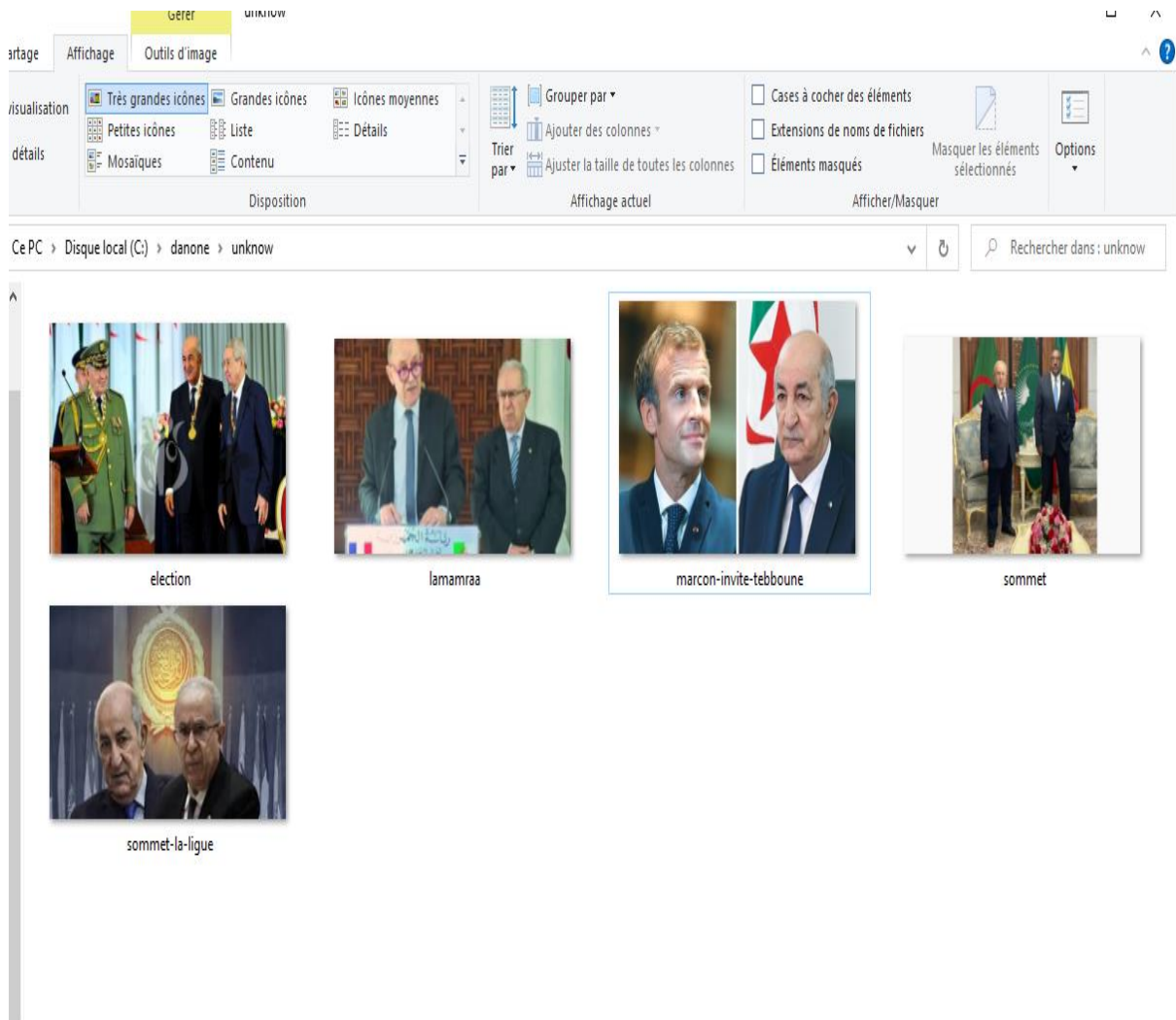


Figure 3.14 Captures de la base de teste.

11. La phase d'analyse de résultat :

Avec une tolérance de 55% déclaré par défaut dans notre code on a obtenu des résultats satisfaisants comme illustrer par la suite :

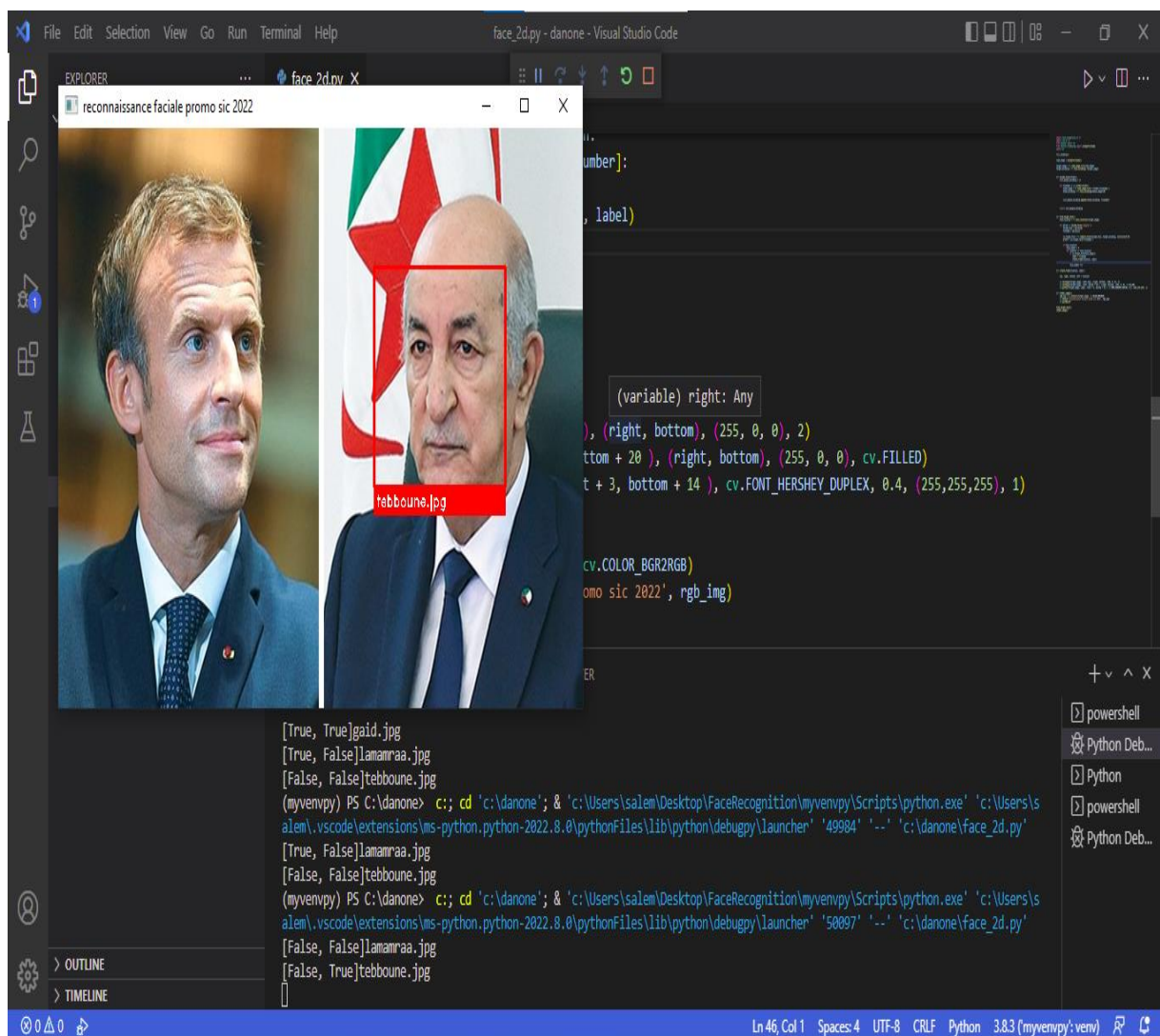


Figure 3.15 capture de la reconnaissance (Tebboune) .

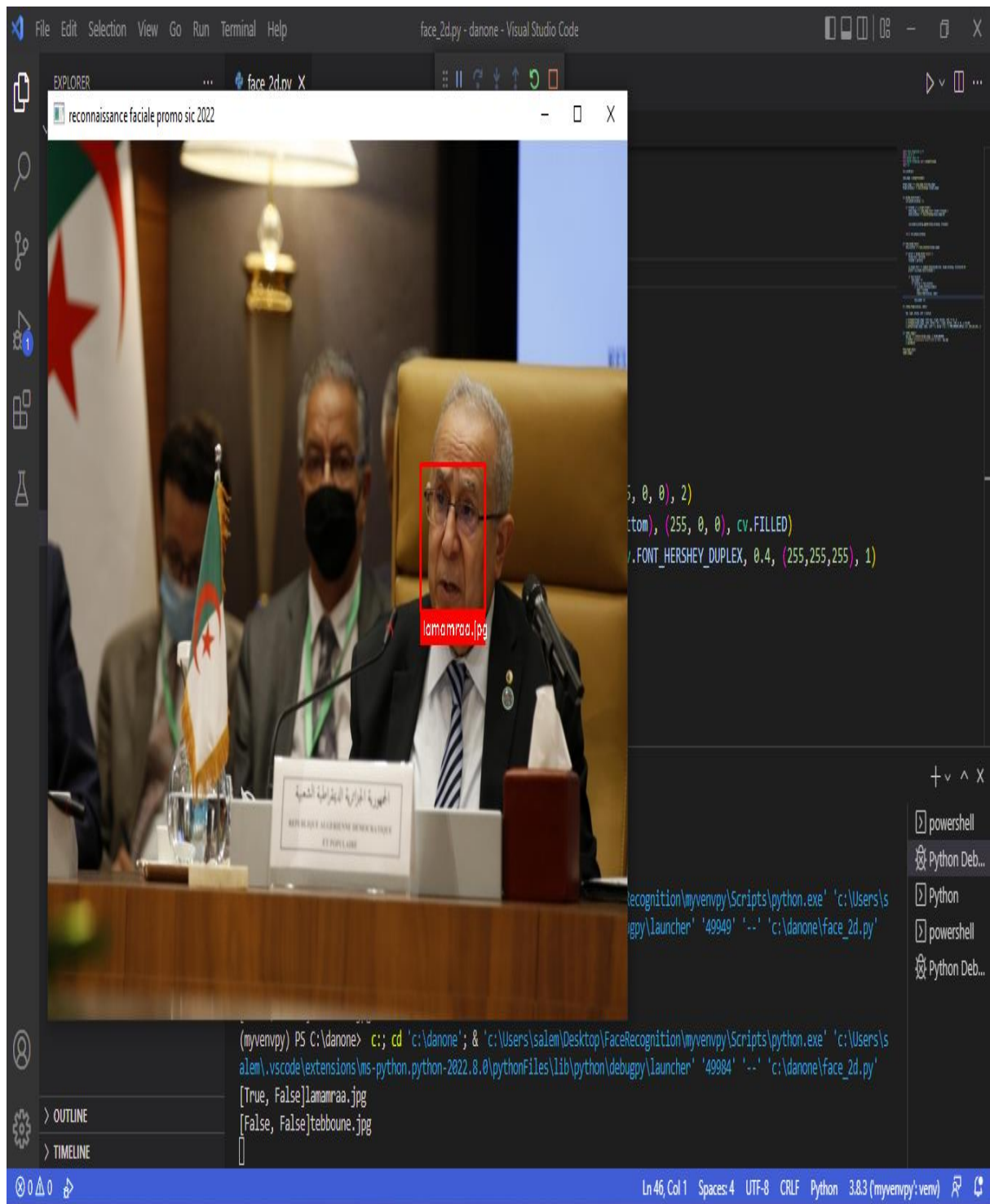
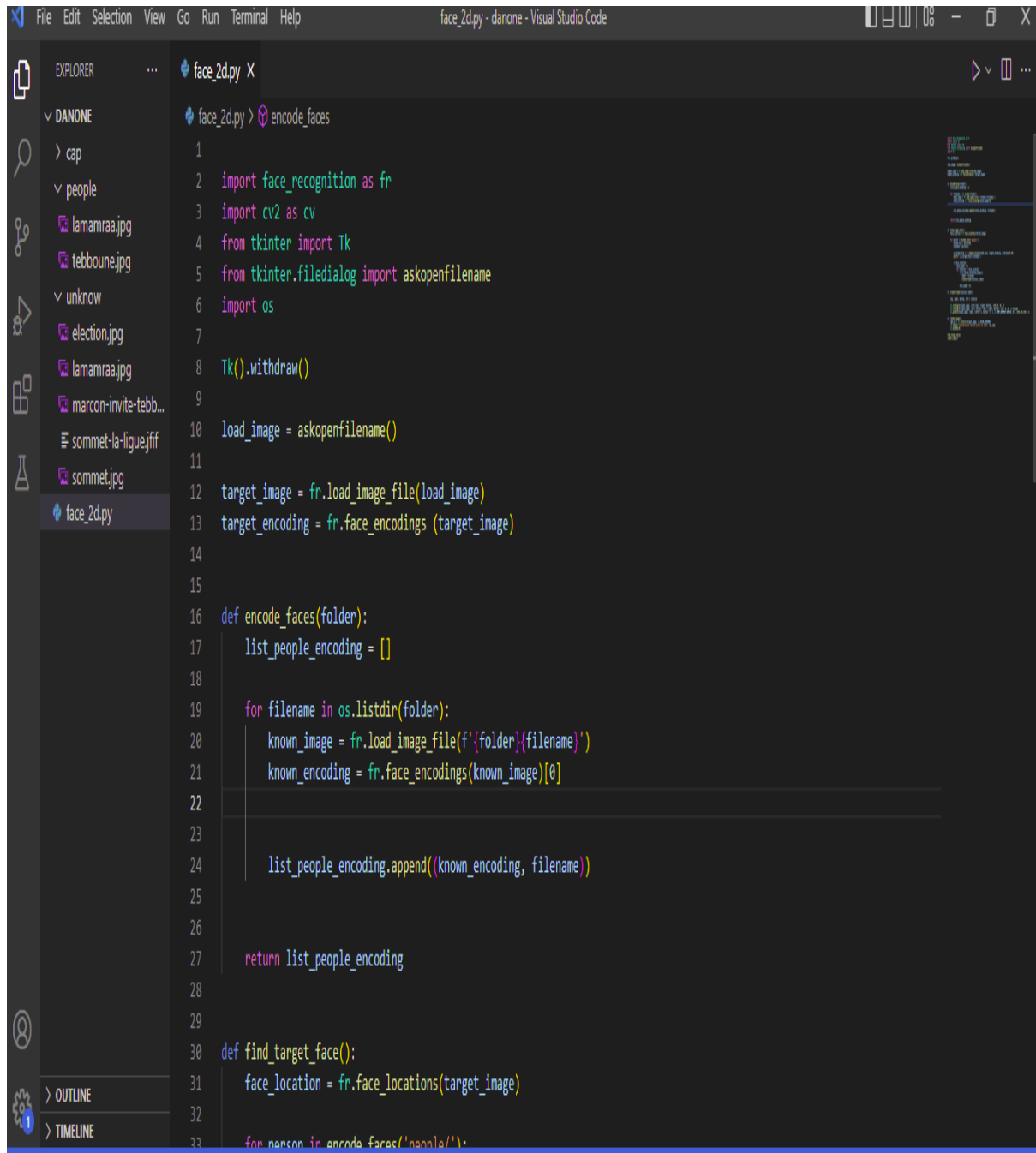


Figure 3.16 capture de la reconnaissance (Lamamra) .

12. Le code :



```
1
2 import face_recognition as fr
3 import cv2 as cv
4 from tkinter import Tk
5 from tkinter.filedialog import askopenfilename
6 import os
7
8 Tk().withdraw()
9
10 load_image = askopenfilename()
11
12 target_image = fr.load_image_file(load_image)
13 target_encoding = fr.face_encodings(target_image)
14
15
16 def encode_faces(folder):
17     list_people_encoding = []
18
19     for filename in os.listdir(folder):
20         known_image = fr.load_image_file(f'{folder}{filename}')
21         known_encoding = fr.face_encodings(known_image)[0]
22
23
24         list_people_encoding.append((known_encoding, filename))
25
26
27     return list_people_encoding
28
29
30 def find_target_face():
31     face_location = fr.face_locations(target_image)
32
33     for person in encode_faces('people/')
```

Figure 3.17 Capture de code.

```

32
33     for person in encode_faces('people/'):
34         encode_face = person[0]
35         filename = person[1]
36
37         is_target_face = fr.compare_faces(encode_face, target_encoding, tolerance=0.55)
38         print(f'{is_target_face}{filename}')
39
40         if face_location:
41             face_number = 0
42             for location in face_location:
43                 if is_target_face[face_number]:
44                     label = filename
45                     create_frame(location, label)
46
47                 face_number += 1
48
49 def create_frame(location, label):
50
51     top, right, bottom, left = location
52
53     cv.rectangle(target_image, (left,top), (right, bottom), (255, 0, 0), 2)
54     cv.rectangle(target_image, (left, bottom + 20 ), (right, bottom), (255, 0, 0), cv.FILLED)
55     cv.putText(target_image, label, (left + 3, bottom + 14 ), cv.FONT_HERSHEY_DUPLEX, 0.4, (255,255,255), 1)
56
57 def render_image():
58     rgb_img = cv.cvtColor(target_image, cv.COLOR_BGR2RGB)
59     cv.imshow('reconnaissance faciale promo sic 2022', rgb_img)
60     cv.waitKey(0)
61
62 find_target_face()
63 render_image()

```

Figure 3.18 Capture de code.

13. Conclusion :

La reconnaissance faciale est un moyen d'identifier ou de confirmer l'identité d'un individu grâce à son visage. Les systèmes de reconnaissance faciale peuvent servir à l'identification de personnes sur des photos, dans des vidéos ou en temps réel. Il existe plusieurs méthodes de reconnaissance mais qui sont très gourmande en ressource surtout en processeur (GPU, TPU) c'est pour ça on choisit l'Algorithme face recognition avec open cv pour la détection et l'analyse de visage. Le logiciel analyse la géométrie du visage, les facteurs clés incluent la distance entre les yeux, la profondeur des orbites, la distance entre le front et le menton, la forme des pommettes, ainsi que le contour des lèvres, des oreilles et du menton. Le but est d'identifier les spécificités de visage. Dans la phase de conversion de l'image en données on n'a pas trouvé de soucis car le processus de capture du visage transforme les informations analogues (un visage) en un ensemble d'informations numériques (les données) selon les caractéristiques du visage de la personne. En fait, l'analyse de votre visage est transformée en formule mathématique. Au final la méthode choisie donne des résultats très satisfaisantes on arrive à reconnaître 80 pour cent des visages introduites dans la base avec un temps record et de même une reconnaissance multiple de visage à condition qu'il faut introduire toutes les images à reconnaître en question déjà dans la base d'apprentissage.

Conclusion générale :

. Aujourd'hui l'identification surpasse la relation de l'Etat avec son citoyen. Des acteurs internationaux, comme l'OACI ou les institutions européennes, jouent eux aussi un rôle prépondérant. Le marché aussi a son mot à dire puisque les industriels développent des relations avec les gouvernements et les entités de standardisations. Toutefois, la décision de recourir à la biométrie repose sur des logiques qui vont au delà d'une rationalité économique. Elle est essentiellement politique en ce sens qu'elle vise à créer au sein de la population le sentiment d'être protégé par l'Etat et ses agences de sécurité dans un contexte d'incertitudes. Ce sentiment contribue fortement à réaffirmer la fonction wébérienne de la politique, celui du monopole de la violence légitime détenu par l'Etat, que les attentats du 11 septembre 2001 avaient défié et effrité⁸⁷. Ce qui a commencé comme une guerre contre le terrorisme a été bascule vers un nouvel agenda sécuritaire et ce qui a commencé par donner du pouvoir aux agences gouvernementales pour combattre le terrorisme a donné plus de pouvoir au gouvernement de manière générale. C'est exactement ce que les opposants à cette technologie invoquent comme argument. Ils craignent le détournement de finalité qui conduirait ainsi les individus à vivre dans une « société de contrôle ». Cette expression développée par les philosophes Gilles Deleuze et Antonio Negri s'inspire de la pensée de Michel Foucault. Ainsi ce type d'organisation de la société, qui s'appuie sur l'évolution technique et le développement des technologies de l'information et de la communication, semble garantir une plus grande marge de manœuvre aux individus et davantage de mobilité. Contrairement aux dispositifs disciplinaires classiques qui procèdent par la coercition, aujourd'hui le mouvement et la liberté de circulation sont les conditions nécessaires à l'exercice d'un pouvoir qui opère désormais par « contrôle continu » de tous les aspects de l'existence et par « communication instantanée ». On remarque à l'heure actuelle une dichotomie qui devrait être dépassée, Bruce Schneier disait que « la sécurité n'est pas liée à la technologie, elle est liée aux risques.

Les références :

- [1] H. A. Rowley, S. Baluja, et T. Kanade, "Neural Network based Face Détection", IEEE Trans, Pattern Anal. Mach, Intell, 23-38, January 1998.
- [2] G.Bencherki et Moustafa. «Thème Implémentation d'un système de reconnaissance de visages à base de PCA», Mémoire de Master, Université Djilali Bounaama, Khemis Miliana, 2018.
- [3] GHALI Ahmed. «Amélioration de la reconnaissance par le visage», Mémoire de Magister, UNIVERSITE MOHAMED BOUDIAF ORAN, 2015.
- [4] Ghouliesbouthaina. Kouidri Yamina. «Etude comparative d'ensemble des descripteurs de texture pour la reconnaissance de visages», Université Kasdi Merbah Ouargla, 2017.
- [5] DJEDI Sara, «Etude comparative de PCA et KPCA associées au SVM en biométrie», Université Mohamed Khi der Biskra, 2012.
- [6] Boudjellal. S, «Détection et identification de personne par méthode biométrique», Mémoire de Magister ,Université Mouloud Mammeri, 2017.
- [7] Franck Davoine, Bouchra Abboud et Van Mô Dang, «Analyse de visages et d'expressions faciales par modèle actif d'apparence», Université de Technologie de Compiègne, France, 2004.
- [8] A.J. O'Toole, H. Abdi, «Low-dimensional representation of faces in higher dimensions of the face space», Opt. Soc. Am. Vol. 10, no. 23, p.405-411, 1993.
- [9] Belahcene Mébarka, «Analyse de texture par ACP», Magister électricité industrie, Université de Batna; Octobre 1994.
- [10] M.Adjout et A.Benaissa, «Fusion de la DCT-PCA et la DCT-LDA appliquée à la reconnaissance de visages», Institut National de formation en Informatique (I.N.I) Oued-smar. Alger, 2007.
- [11] N. c Son VU, «Contributions à la reconnaissance de visages à partir d'une seule image et dans un contexte non-contrôlé», thèse de doctorat. Institut polytechnique de Grenoble, novembre 2010.
- [12] M. Van Wambeke, «Reconnaissance et suivi de visages et implémentation en robotique», Master Ingénieur Civil en Génie Biomédical, Université Catholique de Louvain. 2009/2010.

- [13] R.Gross, J.shi, j.Cohn, «Quo Vadis face Recognition», Third Workshop on Empirical Evaluation Méthode in computer Vision, Décembre, 2001.
- [14] Deng, L., & Yu, D. (2014). Deep Learning: méthodes and applications. *Fondations and Trends® in Signal Processing*, 7(3–4), 197-387.
- [15] Bengio, Y. (2009). Learning deep architectures for AI. *Fondations and trends in Machine Learning*, 2(1), 1-127.
- [16] Bengio, Y., Courville, A., & Vincent, P. (2013). Representation Learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8), 1798-1828.
- [17] Schmid Huber, J. (2015). Deep Learning in neural networks: An overview. *Neural networks*, 61, 85-117.
- [18] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep Learning. *Nature*, 521(7553), 436.
- [19] Arel, I., Rose, D. C., & Karnowski, T. P. (2010). Deep machine Learning-a new frontier in artificial intelligence research. *IEEE computational intelligence magazine*, 5(4), 13-18.
- [20] J. Schmid Huber, “Deep Learning,” *Scholarpedia*, vol. 10, no. 11, p. 32832, 2015.
- [21] Moualek, D. Y. (2017). Deep Learning pour la classification des images (Thèse de doctorat), Université Abou Bakr Belkaid– Tlemcen.
- [22] Y. Bengio, A. Courville, and P. Vincent. Representation Learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 38:1798–1828, 2013
- [23] Hubara, I., Courbariaux, M., Soudry, D., El-Yaniv, R., & Bengio, Y. (2017). Quantized neural networks: Training neural networks with low precision weights and activations. *The Journal of Machine Learning Research*, 18(1), 6869-6898.
- [24] Van den Oord, A., Dieleman, S., & Schrauwen, B. (2013). Deep content-based music recommendation. In *Advances in neural information processing systems* (pp. 2643-2651).
- [25] Collobert, R., & Weston, J. (2008, July). A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning* (pp. 160-167).

- [26] Matsugu, M., Mori, K., Mitari, Y., & Kaneda, Y. (2003). Subject independent facial expression recognition with robust face detection using a convolutional neural network. *Neural Networks*, 16(5-6), 555-559.
- [27] Mikolov, T., Karafiát, M., Burget, L., Černocký, J., & Khudanpur, S. (2010). Recurrent neural network based language model. In *Eleventh annual conference of the international speech communication association*.
- [28] Liu, S., Yang, N., Li, M., & Zhou, M. (2014, June). A recursive recurrent neural network for statistical machine translation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 1491-1500).
- [29] Graves, A., & Jaitly, N. (2014, January). Towards end-to-end speech recognition with recurrent neural networks. In *International conference on machine learning* (pp. 1764-1772).
- [30] Karpathy, A., & Fei-Fei, L. (2015). Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3128-3137).
- [31] Ng, A. Y., & Jordan, M. I. (2002). On discriminative vs. generative classifiers: A comparison of logistique régression and naïve bayes. In *Advances in neural information processing systems* (pp. 841-848).
- [32] Ackley, D. H., Hinton, G. E., & Sejnowski, T. J. (1985). A learning algorithm for Boltzmann machines. *Cognitive science*, 9(1), 147-169.
- [33] Salakhutdinov, R., Mnih, A., & Hinton, G. (2007, June). Restricted Boltzmann machines for collaborative filtering. In *Proceedings of the 24th international conference on Machine learning* (pp. 791-798).
- [34] Hinton, G. E. (2012). A practical guide to training restricted Boltzmann machines. In *Neural networks: Tricks of the trade* (pp. 599-619). Springer, Berlin, Heidelberg.
- [35] Hinton, G. E., Oslander, S., & Teh, Y. W. (2006). A fast Learning algorithm for deep belief nets. *Neural computation*, 18(7), 1527-1554.
- [36] Salakhutdinov, R., & Hinton, G. (2009, April). Deep Boltzmann machines. In *Artificial intelligence and statistics* (pp. 448-455).4
- [37] Srivastava, N., & Salakhutdinov, R. R. (2012). Multimodal Learning with deep Boltzmann machines. In *Advances in neural information processing systems* (pp. 2222-2230).

- [38] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2014). Générateur adversaire nets. In Advances in neural information processing systems (pp. 2672-2680).
- [39] Denton, E. L., Chintala, S., & Fergus, R. (2015). Deep generative image model using a laplacian pyramid of adversarial networks. In Advances in neural information processing systems (pp. 1486-1494).
- [40] Zhang, R., Zhu, J. Y., Isola, P., Geng, X., Lin, A. S., Yue, T., & Efros, A. A. (2017). Real time user-guided image colorisation with learned deep priors. ArXiv preprint arXiv:1705.02999.
- [41] Owens, A., Isola, P., McDermott, J., Torralba, A., Adelson, E. H., & Freeman, W. T. (2016). Visually indicated sounds. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2405-2413).
- [42] Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwan, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. ArXiv preprint arXiv:1406.1078.
- [43] Sze, D., Szegedy, C., Toshev, A., & Anguelov, D. (2014). Scalable Object Detection using deep neural networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2147-2154).
- [44] Graves, A. (2013). Generating sequences with recurrent neural networks. ArXiv preprint arXiv:1308.0850.
- [45] Karpathy, A., & Fei-Fei, L. (2015). Deep visual-semantic alignments for generating image descriptions. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 3128-3137).
- [46] Minh, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., & Riedmiller, M. (2013). Playing Atari with deep reinforcement Learning. ArXiv preprint arXiv:1312.5602.
- [47] Redford, A., Metz, L., & Chintala, S. (2015). Unsupervised representation Learning with deep convolutional generative adversarial networks. ArXiv preprint arXiv:1511.06434.
- [48] Bengio, Y., Laufer, E., Alain, G., & Jasinski, J. (2014, January). Deep generative stochastic networks trainable by backprop. In International Conference on Machine Learning (pp. 226-234).
- [49] Makhzen, A., Shlens, J., Jaitly, N., Goodfellow, I., & Frey, B. (2015). Adversarial autoencoders. ArXiv preprint arXiv:1511.05644.

[50] S.Raschka, Author of Python Machine Learning, researcher applying ML to computational bio .17 aout 2015 : What is the différenceBe tween deep Learning and usuel machine Learning?

[51] J.Da Rolt, PhD, Computer Engineer, Professor 26.12.2017: How is a convolutional neural network able to learn invariant features?

[52] Van den Oord, A., Diele man, S., & Schrauwen, B. (2013). Deep content-based music recommandation. In Advances in neural information processing systems (pp. 2643-2651).

[53] Dauphin, Y. N., Pascanu, R., Gulcehre, C., Cho, K., Gangui, S., & Bengio, Y. (2014). Identifying and Attac King the saddle point problem in high-dimensional non-convexe optimisation. In Advances in neural information processing systems (pp. 2933-2941).

[54] Duchi, J., Hazan, E., & Singer, Y. (2011). Adaptive subgradient méthodes for online Learning and stochastic optimization. Journal of Machine Learning Research, 12(Jul), 2121- 2159.

[55] Dean, J., Corrado, G., Monga, R., Chen, K., Devin, M., Mao, Ng, A. Y. (2012). Large scale distributed deep networks. In Advances in neural information processing systems (pp. 1223-1231).

[56] Clark, L. (2012). Googleartificiel brain learns to find cat videos. Wired UK, www. Wired. Com.

[57] Bennington, J., Socher, R., & Manning, C. (2014, October). Glove: Global vectors for Wordreprésentation. In Proceedings of the 2014 conférence on empirical méthodes in Naturallangage processing (EMNLP) (pp. 1532-1543).

[58] Hinton, G., Srivastava, N., & Swersky, K. (2012). Neural networks for machine Learning lecture 6a overview of mini-batch gradient descent. Cited on, 14, 8.

Webographie :

- [w1] <https://static.bodet-software.com/images/content/FR/badgeuse/biometrique/technologie-biometrique.jpg> consulter 20_04_2022.
- [w2] <https://infodaffaires.com/un-systeme-de-securite-qui-passe-par-la-reconnaissance-oculaire-chez-hyundai/> consulter 23_04_2022.
- [w3] https://fr.123rf.com/photo_39817800_syst%C3%A8me-de-s%C3%A9curit%C3%A9-biom%C3%A9trique-futur-de-dispositif-de-balayage-d-empresnte-digitale.html consulter 14_05_2022.
- [w4] <https://www.dynamique-mag.com/article/astuces-securiser-mots-passe.11140> consulter 24_05_2022.
- [w5] <https://www.developpez.com/actu/239863/La-reconnaissance-faciale-comme-methode-de-securite-est-inutile-pour-4-smartphones-sur-10-d-apres-une-etude/> consulter 11_04_2022.
- [w6] <https://www.alliancevita.org/2022/04/decryptage-de-l-adn-les-scientifiques-font-de-nouvelles-decouvertes-sur-le-genome-humain/> consulter 03_05_2022.
- [w7] <https://fr.wikipedia.org/wiki/Signature> consulter 9_05_2022.
- [w8] https://ru.freepik.com/premium-photo/minimalistic-blue-bars-voice-graphic-on-white-background-equalizer_19500935.htm consulter 14_05_2022.
- [w9] <https://www.journaldunet.fr/business/dictionnaire-du-marketing/1208194-reconnaissance-faciale-definition-et-traduction/> consulter 07_05_2022.
- [w10] [https://www.wedemain.fr/inventer/9-dates-qui-ont-marque-l-essor-de-la-reconnaissance faciale_a4568-](https://www.wedemain.fr/inventer/9-dates-qui-ont-marque-l-essor-de-la-reconnaissance-faciale_a4568-) consulter 14_05_2022.
- [w11] <http://thesis.univ-biskra.dz/944/4/Chap%201%20Syst%C3%A8me%20RV%20sept%2012.pdf> consulter 20_04_2022.
- [w12] <https://penseeartificielle.fr/focus-comment-marche-la-reconnaissance-faciale/> consulter 14_05_2022.

- [w13] https://www.researchgate.net/figure/Exemple-dun-visage-dune-meme-personne-subissant-des-variations-de-pose-hors-plan_fig3_291345615 consulter 17_04_2022.
- [W14] www.fr.wikipedia.org . consulter 21_04_2022.
- [W15] <https://www.futura-sciences.com/tech/dossiers/robotique-presentation-historiquereseaux-neuronaux-31/> consulter 27_04_2022.
- [W16] www.image-net.org consulter 29_04_2022.
- [W17] www.fr.wikiversity.org consulter 28_04_2022.
- [W18] <https://machinelearningmastery.com/gentle-introduction-mini-batch-gradient-descentconfigure-batch-size/c> consulter 11_04_2022.
- [W19] <https://medium.com/deep-learning-turkey/google-colab-free-gpu-tutorial-e113627b9f5d> consulter 04_05_2022.
- [W20] <https://www.python.org/> consulter 15_04_2022.
- [W21] <https://opencv.org/about/> consulter 16_04_2022.
- [W22] <http://www.numpy.org/> consulter 11_04_2022.
- [W23] <https://www.courspython.com/apprendre-numpy.html> consulter 17_04_2022.
- [W24] <https://fr.wikipedia.org/wiki/Keras> consulter 03_04_2022.
- [W25] <https://pandas.pydata.org/> consulter 13_04_2022.

ملخص

التعرف على الوجه هو أحد التقنيات البيومترية، يعتمد على الذكاء الاصطناعي. و مع ازدياد حجم البيانات على الشبكة العالمية، أصبح الاهتمام بتأمين معلوماتها أكثر أهمية، ومن ثم ظهور عدة تقنيات للتعرف على الوجوه.

يركز هذا العمل على التعرف على الوجه ثنائي الأبعاد، حيث يتم تنفيذ نظام التعرف على الوجوه، باستخدام تقنية (CNN). وبهذا المعنى، فإن الخطوة الأولى هي اكتشاف وجود أو عدم وجود وجه في الصورة. ثم استخراج ملامح الوجه وتخزينها في قاعدة البيانات حتى نتمكن من التعرف عليها لاحقاً. والنتائج التي حققناها مشجعة.

Abstract

Facial recognition is one of the biometric technologies, it's based on artificial intelligence. With the increase in the volume of data on the Web, the concern to secure its information becomes more and more important, hence the appearance of several techniques to recognize faces.

This work focuses on 2D facial recognition, where a face recognition system is implemented, using (CNN) technology. In this sense, the first step is to detect the presence or absence of a face in the image. Then extract the facial features and store them in the database so we can recognize them later. The results we have achieved are encouraging.

Résumé

La reconnaissance faciale fait partie des technologies biométriques, elle est basée sur l'intelligence artificielle. Avec l'augmentation de volume des données sur le Web, le souci de sécuriser ses informations devient de plus en plus important, d'où l'apparition de plusieurs techniques pour reconnaître les visages.

Dans ce travail on s'intéresse à la reconnaissance faciale 2D, où un système de reconnaissance de visages est mis en œuvre, en utilisant la technologie de (CNN). En ce sens, la première étape consiste à détecter la présence ou l'absence d'un visage dans l'image. Puis extraire les traits du visage et les stocker dans la base de données afin que nous puissions les reconnaître plus tard. Les résultats que nous avons obtenus sont encourageants.