



MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE
LA RECHERCHE SCIENTIFIQUE
UNIVERSITÉ ABDELHAMID IBN BADIS - MOSTAGANEM

Faculté des Sciences Exactes et de l'Informatique
Département de Mathématiques et d'Informatique
Filière : Informatique

MEMOIRE DE FIN D'ETUDES

Pour l'Obtention du Diplôme de Master en Informatique
Option : **Ingénierie des Systèmes d'Information**

THEME :

**Implémentation d'un système profond
d'identification faciale**

Etudiant(e) : GORINE Abdessamad.

Encadrant(e) : BENTAOUZA Chahinez Mériém.

Dédicaces

Au nom de dieu le clément et le miséricordieux louange à **ALLAH** le tout puissant.

Je dédie ce modeste travail en signe de respect, reconnaissance et de remerciement :

A mes parents :

Sans leurs soutiens et leurs conseils, mes accomplissements n'auraient pas eu lieu, ils ont été derrière moi dans chacun de mes pas tout au long de ma vie, ma plus profonde gratitude leur a exprimé, aucun mot pourrait qualifier l'estime que je leur porte ni le bien qu'ils m'ont fait, apporter et donner.

A ma sœur Sarah, pour leur amour, soutien et leur présence dans ma vie.

A mes grands-parents, que dieu la grande pour nous.

A mes oncles et mes tantes, merci pour votre soutien et vos encouragements.

A mes voisins Nabil Fouzi, Fethi, Helim, Dondkibe, merci pour vos

Remerciements

En tout premier lieu, je remercie le bon Dieu, tout puissant de m'avoir donné la force pour survivre, ainsi que l'audace pour dépasser toutes les difficultés

Je tiens à exprime toute ma reconnaissance à mon Directeur de mémoire madame Chahinez Mériém Bentaouza. Je la remercie de m'avoir encadré, orienté et aidé à mettre en œuvre ce travail, j'aimerais aussi la remercier pour ses précieux conseils qui m'ont permis de mener à bien ce travail. J'ai profité pendant longtemps de savoir et de savoir-faire dont j'ai pu bénéficier au cours de nombreuses discussions.

J'exprime toute ma reconnaissance à mon jury de la faculté de l'informatique et sciences exactes de l'université Abde El Hamid Ibn Badis, trouvent ici l'expression de mes vifs remerciements pour avoir bien voulu juger ce travail.

Je remercie mes très chers parents, qui ont toujours été là pour moi, et qui tiennent une place immense dans mon cœur, Vous m'avez offert plus que tout ce que je pouvais désirer et mériter, et vous jouez un rôle primaire et dominant dans ma vie.

Evidemment toute ce que j'ai acquis grâce à vous, je ne pourrais vous le rendre.

Je remercie très spécialement ma famille, En particulier mes frères Abdelhak, Ishak walid, Mohamed Achraf el dine et ma sœur Sarah pour leur soutien et sa présence dans ma vie

Enfin, un grand merci à Noredinne, Abdellah, Sofian, Djamal Edin, Abderrahmane, Bachir et Sabrina pour leur sincères amitié et confiance, leur soutien inconditionnel et leur encouragement, et à qui je dois ma reconnaissance mon attachement.

J'adresse mes sincères remerciement a tous les intervenants et toutes les

Liste des figures

Figure N°	Titre de la figure	Page
Figure I.1.	Ensemble du processus de reconnaissance d'un visage	5
Figure I.2.	Technique de reconnaissance faciale	6
Figure II .1.	Principe de Deep Face	11
Figure II.2.	Architecture de CNN	14
Figure III.1.	Convolution sur image	20
Figure III.2.	Principe de Pooling	21
Figure III.3.	Principe Unités Rectifié Linéaire (ReLU)	21
Figure III.4.	Apprentissage profond sur visage	22
Figure III.5.	Les couches entièrement connectées	23
Figure III.6.	Organigramme de système d'identification	24
Figure IV.1.	Présente la bibliothèque plus utilisée dans trois mois	28
Figure IV.2.	Architecture Inception-v3 layer model	29
Figure IV.3.	Résultat de détection plusieurs face sur un image	30
Figure IV.4.	Détecter les repères faciaux 2D en images	31
Figure IV.5.	Détecter les repères faciaux 3D en images	31
Figure IV.6.	Training le modale	32
Figure IV.7.	Image classifier et résultat de classification	32
Figure IV.8.	Les architectures de P-Net, R-Net, et O-Net	33
Figure IV.9.	Exemple sur la génération de fichier json présente le visage	34

Figure IV.16.	Résultat de l'entropie croisée et la validation	39
Figure IV.17.	Résultat de la précision et la validation	39

Liste des tableaux

Tableau N°	Titre du tableau	Page
Tableau III.1.	Taux de classification des identifications humaine.	18

Liste des abréviations

Abréviation	Expression Complète
ADL	Analyse Discriminantes Linéaire
ACP	Analyse en Composantes Principales
RNA	Réseaux de Neurones
CNN	Réseaux de Neurones à Convolution
SVR	Support Vector Regression
3D	3 Dimensions (dans l'espace)
SFC	Social Face Classification
ReLU	Unités Rectifié Linéaire
LBP	Local Binary Patterns, Motifs Binaries Locaux
LFW	Labeled Faces in the Wild

Résumé

Ce mini-projet est une recherche bibliographique sur les systèmes profonds de reconnaissance faciale. La reconnaissance de visage consiste à déterminer l'identité d'une personne à partir d'une image de son visage. Mais en réalité, toutes les images captées ne sont pas de face et peuvent être de profil. On peut citer l'exemple de Deep Face qui est un système de reconnaissance faciale à apprentissage profond créé par un groupe de recherche sur Facebook. Il identifie les visages humains dans les images numériques. Donc, il est nécessaire de relever tous les travaux de recherche dans ce domaine.

Nous avons fait des recherches sur le Deep face et on va développer un système avec Python destinée pour la reconnaissance des visages, de l'âge et de genre ainsi que des célébrités., Cette application est basée sur le Deep Learning, en utilisant des modèles inclus dans le Framework Tensorflow qui ont montré des résultats intéressants au niveau de la reconnaissance ainsi qu'un taux d'erreur petit.

La méthode de Deep Face donne des résultats intéressants au niveau de la reconnaissance surtout pour une base d'images grand.

Mots clés :

Deep Learning, Deep face, Reconnaissance faciale, Apprentissage profond, Réseaux de neurones convolution, Profil.

Table des matières

Introduction générale.....	8
I. Reconnaissance faciale.....	9
I.1 Introduction	9
I.2 La reconnaissance faciale	9
I.3 Principes de la reconnaissance de visages.....	10
I.4 Techniques de reconnaissance faciale	11
I.4.1 Méthodes globales.....	11
I.4.2 Méthodes locales	12
I.4.3 Méthodes hybrides	13
I.4.4 Réseaux de neurones.....	13
I.5 Avantage et inconvénient de la reconnaissance de visage.....	13
I.6 Conclusion	14
II. Deep Face.....	15
II.1 Introduction	15
II.2 Définition.....	15
II.3 Principe.....	16
II.4 Alignement du visage / Frontalisation.....	17
II.4.1 Alignement 2D	17
II.4.2 Alignement 3D	17
II.5 Architecteur de CNN	18
II.6 Entraînement.....	19
II.7 Vérification du visage.....	19
II.8 Résultat de deep face	20
II.9 Conclusion	21
III. Conception	22
III.1 Introduction.....	22
III.2 Le Synthèse.....	22

III.6	Softmax	28
III.7	Organigramme de Système	29
III.8	Conclusion	30
IV.	Réalisation.....	31
IV.1	Introduction	31
IV.2	Environnement matériel et logiciel.....	31
IV.2.1	Les ressources physiques suivant.....	31
IV.2.2	Pour ce qui est côté logiciel :	31
IV.3	Langage de programmation	31
IV.4	Les bibliothèques utilisées.....	32
IV.4.1	TensorFlow	32
IV.4.2	Dlib.....	33
IV.4.3	Comment ça marche TensorFlow	34
IV.4.4	Le modèle inception-v3.....	34
IV.5	Système iKnow you.....	35
IV.5.1	Détection	35
IV.5.2	Présentation de visage	35
IV.5.3	Classification	36
IV.5.4	Capture sur l'interface	40
IV.5.5	Visualisation de donnée.....	42
IV.6	Conclusion.....	45
	Conclusion Générale	46
	Bibliographies	47

Introduction générale

La reconnaissance des objets est parmi les sujets les plus importants en vision par ordinateur, il a subi une immense évolution pendant des siècles, en revanche les dernières années l'intelligence artificielle a connu l'apparition de Deep Learning, et grâce aux efforts des chercheurs le Deep Learning connaît un grand succès, ses applications ont touché des domaines différents, comme la robotique, l'industrie, l'automobile ...

Dans cette recherche, on a pris l'initiative de développer un système profond d'identification faciale de en utilisant le Deep Learning qui est l'objet de ce projet de fin d'études.

En fait, il s'agit du développement et de l'implémentation d'un système de reconnaissance faciale en utilisant le Deep Learning.

Ce document comporte quatre chapitres principaux :

-La reconnaissance faciale : dans ce chapitre nous donnons la définition du système reconnaissance faciale pour découvrir les techniques existantes, je suis intéressé à la méthode la plus utilisée dans ce domaine à savoir : la méthode Réseaux de neurones pour la reconnaissance.

-Le système de Deep Face en utilisant les réseaux de neurones convolution : Dans ce chapitre on abordera les démarches suivies durant pour le développement de cette application, et on finira par résultat de Deep Face sur des bases d'images.

- Conception : au début de ce chapitre nous présentons l'état de l'art de la reconnaissance pour découvrir les techniques existantes, et nous donnons des généralités sur la méthode de Deep Face puis nous explicitons la méthodologie que nous avons suivie pour appliquer cette

I.Reconnaissance faciale

I.1 Introduction

La détection de visage dans l'image est un traitement indispensable et crucial avant la phase de reconnaissance. En effet, le processus de reconnaissance de visages ne pourra jamais devenir intégralement automatique s'il n'a pas été précédé par détection efficace [6].

La reconnaissance des visages dans des images sans contrainte est à la pointe la révolution de la perception algorithmique. Le social et implications culturelles des technologies de reconnaissance faciale sont loin, mais l'écart de performance actuel dans ce domaine entre les machines et le système visuel humain sert comme un tampon de devoir faire face à ces implications. Nous présentons un système (Deep Face) qui a fermé la majorité de l'écart restant dans le référentiel le plus populaire dans la reconnaissance faciale sans contrainte, et est maintenant à la limite de la précision au niveau humain [8]. Il est formé sur un grand ensemble de données de visages acquis d'une population très différente de l'un utilisé pour construire les repères d'évaluation, et il est capable de surpasser les systèmes existants avec seulement très minime adaptation. De plus, le système produit une représentation du visage compact, en contraste absolu avec le changement vers des dizaines de milliers de fonctionnalités d'apparence dans d'autres systèmes récents [1].

I.2 La reconnaissance faciale

La reconnaissance faciale s'inscrit dans le domaine plus vaste de la vision par ordinateur, qui part du constat que le sens le plus utilisé par l'homme est la vue. Dès lors, il peut s'avérer très utile de donner des « yeux » à son ordinateur : ainsi, il peut devenir capable de remplacer ceux de l'homme pour des tâches répétitives telles que la reconnaissance de nombreux visages, ou demandant des calculs conséquents comme le lissage d'une image ou l'augmentation de sa netteté.

I.3 Principes de la reconnaissance de visages

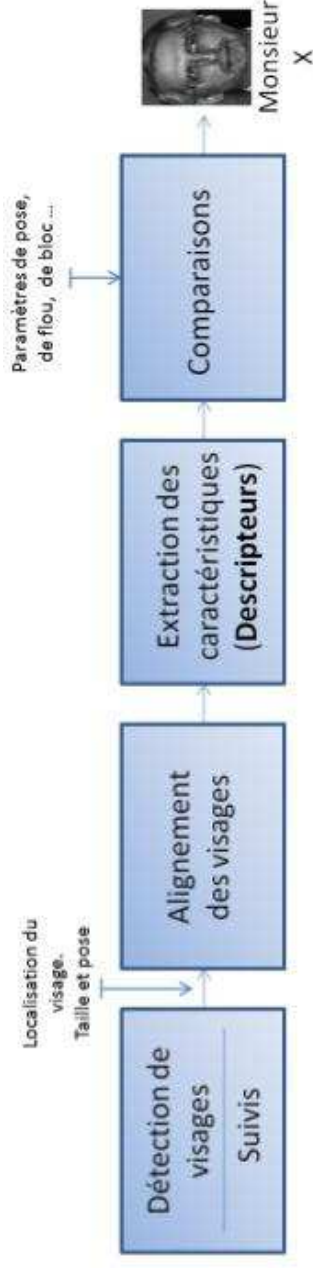


Figure I.1. Ensemble du processus de reconnaissance d'un visage.

Avant d'être capable de reconnaître une personne sur une image ou sur une vidéo, plusieurs étapes de traitement sont nécessaires. Dans les grandes lignes, on peut découper un processus de reconnaissance en quatre étapes principales qui sont présentées sur la figure suivante :

La détection (ou le suivi) du visage d'une personne dans une image ou une vidéo : Cette étape consiste à dire si oui ou non il y a un visage sur une image (ou une personne), combien il y en a et où ils sont situés dans l'image.

L'alignement du visage : Cette étape consiste à permettre une normalisation de l'image de visage détecté afin de pouvoir la comparer avec les autres images de visages de la base qui n'ont pas forcément la même taille ni subi le même éclairage. Cela nécessite donc la détection de certains traits du visage (généralement, les yeux, le nez et la bouche) et la normalisation des distances existantes entre ces traits. Cela suppose également de connaître la pose du visage. L'étape de normalisation d'illumination nécessite également un traitement particulier [2].

L'extraction des caractéristiques du visage : Une fois l'alignement réalisé, il s'agit d'extraire les caractéristiques les plus pertinentes d'un visage pour permettre son identification et ne pas le confondre avec celui d'un autre individu. Le vecteur de caractéristiques ainsi associé à un

I.4 Techniques de reconnaissance faciale

Plusieurs méthodes de reconnaissance de visage ont été proposées ces dernières années, suivant deux axes : la reconnaissance à partir d'image fixes et la reconnaissance à partir de séquence d'image (vidéo). De manière générale, les méthodes de reconnaissance de visage peuvent être divisées en trois groupes : des méthodes globales, des méthodes locales, des méthodes hybrides, réseaux de neurones et modèles cachés de Markov.

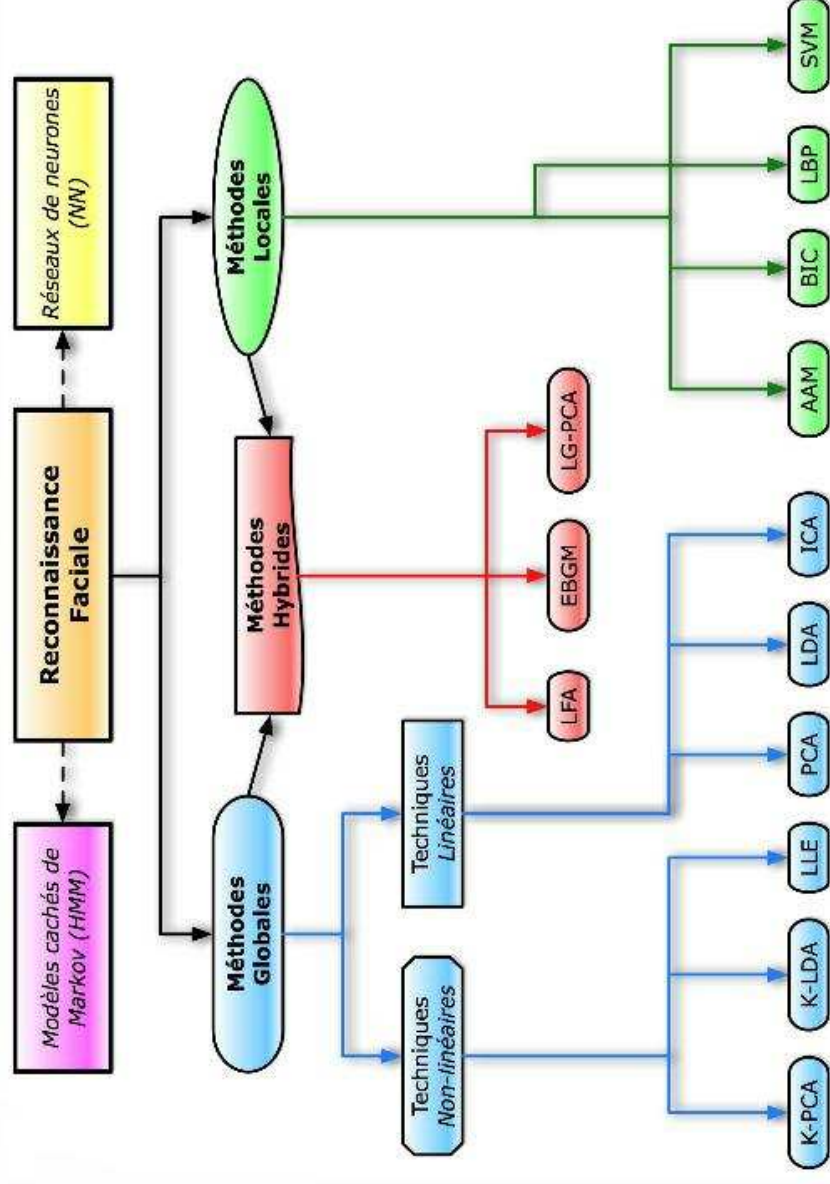


Figure I.2. Technique de reconnaissance faciale.

I.4.1 Méthodes globales

Les méthodes globales prennent l'image du visage comme un tout et utilisent des

très sensibles aux variations d'éclairément, de pose et d'expression faciale [1]. Parmi les algorithmes les plus appliqués dans cette méthode, on cite :

L'algorithme de référence de la reconnaissance de visage « ACP ». C'est une méthode de réduction d'espace connue aussi sous le nom de transformée de Karhunen-Loeve ou Analyse en Composantes Principales. Elle a été utilisée par Turk et Pentland dans Eigenfaces [4], ou la notion de « Eigenface » a été introduite pour exprimer les différents vecteurs propres de la base d'image de visage basée sur une transformation linéaire dans un espace de dimension réduite qui maximise la variance des projections des échantillons originaux [3].

L'Analyse Discriminante Linéaire (ADL), appelée également « Fisher Linear Discriminant Analysis », est aussi parmi l'une des méthodes les plus utilisées pour la reconnaissance de visages. Elle utilise le critère de réduction qui se base sur la notion de séparabilité des données par classe. Elle a été appliquée aux visages en 1996 [5].

I.4.2 Méthodes locales

On les appelle aussi les méthodes à traits, géométriques, à caractéristiques locales ou analytiques. Ce type consiste à appliquer des transformations en des endroits spécifiques de l'image, précisément ces méthodes extraient les caractéristiques locales de visage comme les yeux, le nez et la bouche, puis utilisent leur géométrie et/ou l'apparence comme donnée d'entrée du classificateur.

L'immense avantage des méthodes locales de reconnaissance de visages est qu'elles peuvent modéliser facilement les variations de pose, d'illumination ou encore d'expression que peut subir un visage. Cependant, elles nécessitent souvent le placement manuel de nombreux points d'intérêts pour une bonne précision [5].

Elles peuvent être divisées en deux catégories : l'approche géométrique et l'approche basée sur les graphes.

I.4.3 Méthodes hybrides

Un système de reconnaissance utilise les caractéristiques locales et la région globale du visage pour le reconnaître comme le fait le système de perception humaine. Ces méthodes peuvent potentiellement offrir un meilleur taux de reconnaissance que les deux types de méthodes précédentes.

I.4.4 Réseaux de neurones

Cette technique envisagée utilise des réseaux de neurones comme engin d'apprentissage et de reconnaissance. Lorsqu'on commence, une image brute (ou prétraitée) de dimension fixes constitue habituellement la source d'entrée des réseaux. Les dimensions doivent être établies au préalable car le nombre de neurones sur la couche d'entrée en dépend [2].

Plus les dimensions de l'image sont élevées, plus la complexité et le temps d'apprentissage augmentent. En effet, une image de dimension 130*150 pixels, 19500 neurones seront sur la couche d'entrée, ce qui est énorme. L'apprentissage efficace d'un tel réseau est également douteux [6].

Le nombre de sorties du réseau dépend par ailleurs directement de la quantité d'individus à discriminer. Il est donc évident qu'un apprentissage incrémental (avec de nouveaux individus et non de nouveaux exemples) sera difficile et requerra des ajustements directs à l'architecture de réseau [6].

Certains auteurs ont par ailleurs de variantes de la technique de base en modifiant les données d'entrée. Le coefficient de projections d'images dans un espace des visages (EigenFaces) peut par exemple être utilisés comme source d'information.

Les informations peuvent également être acheminées au réseau : c'est le cas par exemple de la concaténation d'une image prétraitée (dimensions 30*40 pixels) et de l'échantillonnage d'histogrammes de couleurs RGB [19].

- Changement d'illumination.
- Expression faciale.
- Présence ou absence des composants structurales, telles que : la barbe, la moustache et les lunettes.
- Les vrais jumeaux qui ont le même indicatif d'ADN.

I.6 Conclusion

La reconnaissance de visages humains est considérée actuellement comme l'une des tâches les plus importantes pour relever le défi en l'identification des personnes. La facilité et la précision avec lesquelles nous pouvons identifier nos amis comme Facebook et nos ennemis comme dans l'air porte, même dans des conditions défavorables, sont des capacités des plus étonnantes du système visuel humain, le but de la reconnaissance des visages est de concevoir des systèmes informatiques capables d'égaliser les êtres humains dans ce domaine.

Dans ce chapitre, nous avons passé en revue la reconnaissance facile, son principe ainsi ses techniques en parlent brièvement sur le Deep Face qui aura plus détails dans le chapitre qui suit.

II. Deep Face

II.1 Introduction

La reconnaissance des visages dans des images non contraintes est à la pointe de la révolution de la perception algorithmique. Le monde et implications culturelles des technologies de reconnaissance faciale sont loin, mais l'écart de performance actuel dans ce domaine entre le machines et le système visuel humain sert comme un tampon de devoir faire face à ces implications [1].

Nous présentons un système (Deep Face) qui a comblé la majorité de l'écart restant dans le référence le plus populaire dans la reconnaissance faciale sans contrainte, et est maintenant à la limite de la précision au niveau humain. Il est formé sur un grand ensemble de données de visages acquis d'une population très différente de l'un utilisé pour construire les repères d'évaluation, et il est capable de surpasser les systèmes existants avec seulement très minime adaptation. De plus, le système produit une représentation de face compacte, en contraste absolu avec le passage vers des dizaines de milliers de traits d'apparence dans d'autres systèmes récents [8]

Cela fait plusieurs années que Facebook propose un module de reconnaissance faciale. Il permet notamment de tagger des personnes sur des photos afin de les partager plus efficacement ensuite. Avec le projet Deep Face, le réseau social fait un pas de plus dans ce domaine [8].

II.2 Définition

Deep Face est profonde Apprendre la reconnaissance faciale. Le système de proposé diffère de la majorité des contributions sur le terrain en ce qu'il utilise l'apprentissage en profondeur (Deep learning) cadre au lieu de caractéristiques bien conçues. DL est

II.3 Principe

Les versions alignées existantes de plusieurs bases de données de visages aident à améliorer les algorithmes de reconnaissance en fournissant une entrée normalisée. Des méthodes récentes ont montré des moyens efficaces qui compensent ces difficultés en utilisant des techniques d'alignement sophistiquées. Bien que l'alignement soit largement utilisé, aucune solution physiquement correcte complète n'est actuellement présente dans le contexte de la vérification faciale sans contrainte. Les modèles 3D sont tombés en disgrâce ces dernières années, en particulier dans des environnements non contraints [9].

L'entrée attendue est l'image d'un visage (c'est-à-dire qu'il ne recherche pas les visages dans les images, les visages doivent déjà être extraits par une méthode différente).

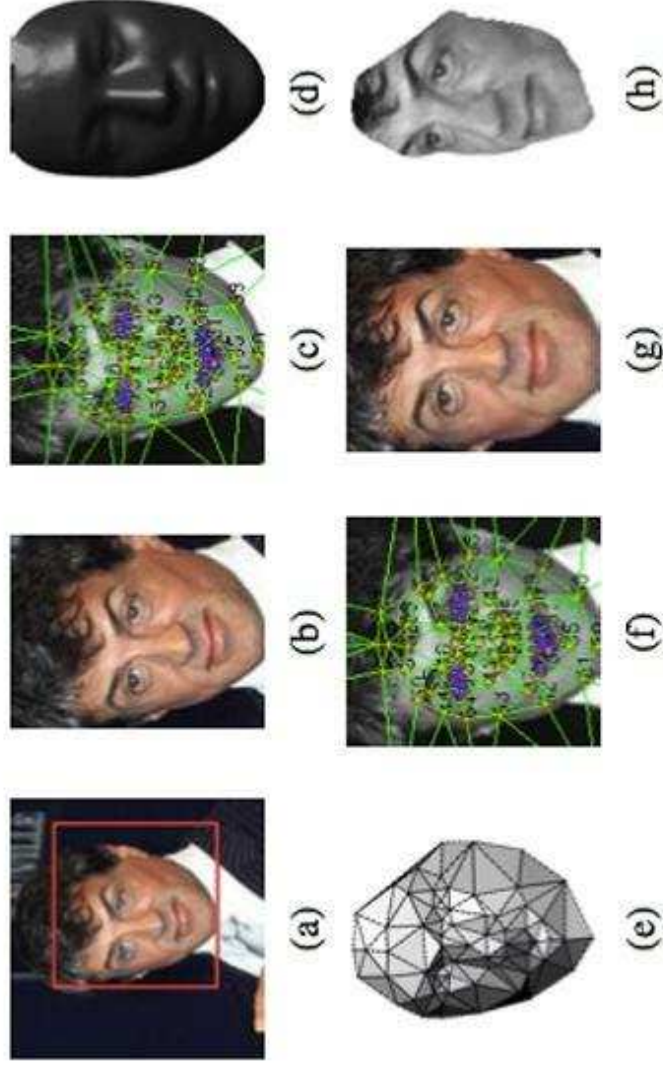


FIGURE II.1. PRINCIPE DE DEEP FACE.

h) Une nouvelle vue générée par le modèle 3D.

II.4 Alignement du visage / Frontalisation

Cible de cette étape : Débarrassez-vous des variations dans les images de visage, de sorte que chaque visage semble regarder droit dans l'appareil photo ("frontalisé").

II.4.1 Alignement 2D

Il commence son processus d'alignement en détectant 6 points de repères à l'intérieur de la culture de détection, centrés au centre des yeux, de la pointe du nez et de la bouche, comme illustré à la figure 3-1 (a).

Cette méthode d'alignement de la figure 3-1 (b) est similaire à celle utilisée dans LFW-a, qui a été fréquemment utilisée pour améliorer la précision de la reconnaissance [4].

- ❖ Ils recherchent des repères (points de repères) sur le visage.
 - Ils utilisent SVR (caractéristiques : LBPs) pour cela.
 - Après chaque application du SVR, les repères localisés sont utilisés pour transformer / normaliser le visage. Ensuite, le SVR est à nouveau appliqué. En faisant cela, les emplacements des repères sont progressivement affinés.
- ❖ Ils utilisent les repères détectés pour normaliser les images de visage (via l'échelle, la rotation et la traduction).

II.4.2 Alignement 3D

- ❖ L'alignement 2D permet de normaliser les variations dans le plan 2D, pas les variations hors plan (par exemple, voir cette face depuis son côté gauche / droit). Pour normaliser les variations hors plan, ils ont besoin d'une transformation 3D.
- ❖ Ils détectent 67 points de repère supplémentaires sur les faces (à nouveau via SVR).

II.5 Architecteur de CNN

Ces dernières années, la littérature sur la vision par ordinateur a attiré de nombreux efforts de recherche en ingénierie des descripteurs. Ces descripteurs, lorsqu'ils sont appliqués à la reconnaissance faciale, utilisent généralement le même opérateur pour tous les emplacements de l'image faciale. Récemment, à mesure que davantage de données devenaient disponibles, les méthodes basées sur l'apprentissage ont commencé à surpasser les fonctionnalités conçues, car elles peuvent découvrir et optimiser les fonctionnalités pour la tâche spécifique en question [4].

Facebook indique que son système utilise neuf réseaux de neurones simulés pour apprendre à reconnaître les modèles d'après une grande quantité de données.

Le CNN reçoit les images de visage frontalisesées (152x152, RVB).

Il applique ensuite les étapes suivantes :

- Convolution, 32 filtres, 11x11, ReLU (-> 32x142x142, CxHxW)
- Max pooling sur 3x3, foulée 2 (-> 32x71x71)
- Convolution, 16 filtres, 9x9, ReLU (-> 16x63x63)
- Convolution locale, 16 filtres, 9x9, ReLU (-> 16x55x55)
- Convolution locale, 16 filtres, 7x7, ReLU (-> 16x25x25)
- Convolution locale, 16 filtres, 5x5, ReLU (-> 16x21x21)
- Entièrement connecté, 4096, ReLU
- Entièrement connecté, 4030, Softmax

Les convolutions locales utilisent un ensemble différent de poids appris à chaque "pixel" (tandis qu'une convolution normale utilise le même ensemble de poids à tous les emplacements)

d'apprentissage. De plus, le vecteur entier est normalisé L2. Le but de cette étape est de rendre le réseau moins sensible aux changements d'illumination [4].

L'ensemble du réseau a environ 120 millions de paramètres.

Visualisation de l'architecture :

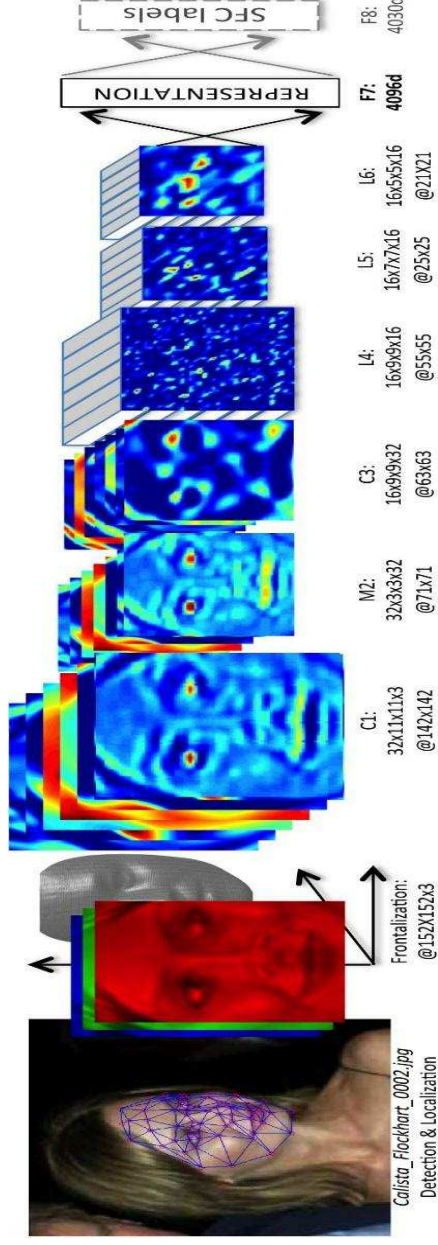


Figure II.2. Architecture de CNN.

II.6 Entraînement

Pour appliquer le deep face il faut l'entraîner pour obtenir un bon résultat, il faut prises le type de l'image et la taille, il ça marche comme suite :

Le réseau reçoit des images, chacune montrant un visage, et est entraîné à classer l'identité du visage.

Ils utilisent l'entropie croisée comme leur perte.

II.7 Vérification du visage

Afin de savoir si deux images de visages montrent la même personne, ils essaient trois méthodes différentes [4].

Chacun de ceux-ci repose sur le vecteur extrait par la première couche entièrement

- c) Réseau siamois. Signifie ici simplement que la distance absolue entre f_1 et f_2 est calculée ($|f_1 - f_2|$), chaque composante est pondérée par un poids appris et ensuite la somme des composantes est calculée. Si le résultat est supérieur à un seuil, les visages sont considérés comme montrant la même personne.

II.8 Résultat de deep face

Ils forment leur réseau sur l'ensemble de données SFC (Social Face Classification). Cela semble être un ensemble de données interne à Facebook (c'est-à-dire non public) avec 4,4 millions de visages de 4k personnes [4].

Lorsqu'il est appliqué à l'ensemble de données LFW :

- Reconnaissance des visages ("quelle personne est montrée dans l'image") (apparemment ils ont recyclé le modèle entier sur LFW pour cette tâche) :
 1. SVM simple avec LBP (c'est-à-dire pas leur réseau) : précision moyenne de 91,4%.
 2. Leur modèle, avec frontalisation, avec alignement 2D : aucune valeur.
 3. Leur modèle, pas de frontalisation (alignement 2D seulement) : précision moyenne de 94,3%.
 4. Leur modèle, pas de frontalisation, pas d'alignement 2D : 87,9% de précision moyenne.
- Vérification de visage (deux images -> même / pas la même personne) (apparemment également formé sur LFW Clair) :
 5. Méthode 1 (produit intérieur + seuil) : précision moyenne de 95,92%.
 6. Méthode 2 (vecteur $X^2 + SVM$) : 97,00% d'acuité moyenne.
 7. Méthode 3 (siamese) : Apparemment 96,17% de précision seule, et 97,25%

II.9 Conclusion

Un classificateur de visage idéal reconnaîtrait les visages avec une précision qui ne correspond qu'aux humains. Le descripteur de visage sous-jacent doit être invariant pour la pose, l'illumination, l'expression et la qualité de l'image. De plus, des descripteurs courts sont préférables et, si possible, des caractéristiques éparses. Certes, le temps de calcul rapide est également une préoccupation. Je pense que ma recherche, qui va basé sur la tendance récente consistant à utiliser davantage de fonctionnalités et à utiliser une technique d'apprentissage profond plus puissante, a permis de relever ce défi et de combler la grande majorité de cet écart de performance. Le recherche démontre que le couplage d'un alignement basé sur un modèle 3D avec méthode de réseaux de neurone de grande capacité peut effectivement apprendre de nombreux exemples pour surmonter avantage de cette méthode, J'explique la méthode qui je veux choisir dans le chapitre suivant.

III. Conception

III.1 Introduction

Comme on a pu le voir dans le chapitre 1, jusqu'à présent peu de travaux de recherche ont été menés sur ce type d'images. Par ailleurs, peu d'entre eux s'intéressent aux images présentant de multiples artefacts. Pourtant, il est rare qu'une image issue d'un système d'identification ne présente qu'un seul artefact. Dans l'idéal, il faudrait que toutes les dégradations générées à un instant t durant le processus d'acquisition (i.e. les problèmes d'illumination, le flou, l'effet de bloc, les problèmes de résolution, etc....) puissent être maîtrisés en même temps. Dans, les auteurs proposent une architecture basée sur l'extraction de caractéristiques locales du visage issues d'images statiques ou de vidéos particulièrement adaptées aux applications de vidéo-surveillance car robustes aux variations de pose, d'illumination et d'expression. Néanmoins, les auteurs ne traitent ni du flou ni des effets de blocs explicitement. De plus, leur méthode est adaptée pour un protocole d'évaluation dont le but est de dire si deux images de visages qui n'ont jamais été vues appartiennent ou non à une même personne. C'est donc un travail de classification binaire : soit la paire d'images appartient à la même personne, soit non. Pour l'identification d'une personne, la tâche est plus difficile dans la mesure où il s'agit de dire si une personne se trouve dans la Base de données et si oui, qui est-ce ? Le travail d'identification est d'autant plus complexe que l'on ne dispose dans la Base de données que d'une seule image par personne prise dans de bonnes conditions d'acquisition [4].

III.2 Le Synthèse

Après les recherche qui je a fait dans le chapitre 1 et 2, je trouvé plusieurs méthodes proposées pour développer système de reconnaissance facial et permet les méthodes qui je vais consulter, il est présent dans le tableau ci-dessus qui présent le taux de classification des

		faciales et signaux physiologiques		
H.Ouamane et M.Benatia	2012	Identification de Reconnaissance Faciale avec des expression	Séparation à veste marge	94,30%
Matthew Zeiler, Rob Fergus	2013	ZF net pour Visualizing and Understanding Convolutional Neural Networks	CNN	88,8 %
Andrej Karpathy	2014	generate natural language descriptions of different image regions	CNN et RNN	92,4 %
Andrej Karpathy's	2015	Microsoft ResNet	152 layer CNN	97,4 %
Groupe de Facebook	2017	Identification de visage dans Facebook pour le Tage (deep face)	CNN Alignement 3D	97,35%

Tableau III.1.1. Taux de classification des identifications humaine.

III.3 La méthode choisie

Ce que ma intéresse à mon travail c'est le Deep Face, Dans la reconnaissance de visage moderne, le pipeline classique se compose de quatre étapes : détecter -> aligner -> représenter-> classifier.

Nous revisitons à la fois l'étape d'alignement et l'étape de représentation en utilisant une modélisation de face 3D explicite afin d'appliquer une transformation affine par morceaux, et dériver un visage représentation à partir d'un réseau neuronal profond à neuf couches. Ce réseau profond implique plus de 120 millions de paramètres en utilisant plusieurs couches connectées localement sans partage de poids, plutôt que les couches convolution elles standard. Ainsi il l'a formé sur le plus grand ensemble de données faciales à ce jour, un jeu de données d'identité étiquetée de quatre millions d'images faciales appartenant à plus de 4000 identités. Les

du point M. Une transformation affine s'obtient en combinant une transformation linéaire du vecteur \overrightarrow{OM} avec une translation :

$$\overrightarrow{OM'} = f(\overrightarrow{OM}) + \vec{b}$$

Ce qui donne sous forme matricielle :

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} + \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix}$$

Les transformations affines sont utilisées pour positionner et orienter des solides dans l'espace, pour les dimensionner ou les déformer, ou bien pour effectuer un changement de repère [10].

III.5 Réseaux de neurones à convolution

Les réseaux de neurones à convolution sont impliqués. Aussi appelés CNN ou ConvNets, ils constituent les fers de lance de l'Apprentissage profond. et on a bien détail dans le chapitre président.

III.5.1 Caractéristique

Le CNN compare les images fragment par fragment. Les fragments qu'il recherche sont appelés les caractéristiques. En trouvant des caractéristiques approximatives qui se ressemblent à peu près dans 2 images différentes, le CNN est bien meilleur à détecter des similitudes que par une comparaison entière image à image [11].

III.5.2 Convolution

En calculant dans toute l'image si une caractéristique est présente, nous faisons un filtrage. Les mathématiques que nous utilisons pour réaliser cette opération sont appelés une convolution, de laquelle les réseaux de neurones à convolution tiennent leur nom.

Les mathématiques derrière le principe de convolution ne sont pas bien complexes. Pour

Si tous les pixels dans une caractéristique correspondent, alors leur addition puis leur division par le nombre total de pixels donne 1. De la même manière, si aucun des pixels de la caractéristique ne correspond à la sous-partie de l'image, alors la réponse est -1 [11].

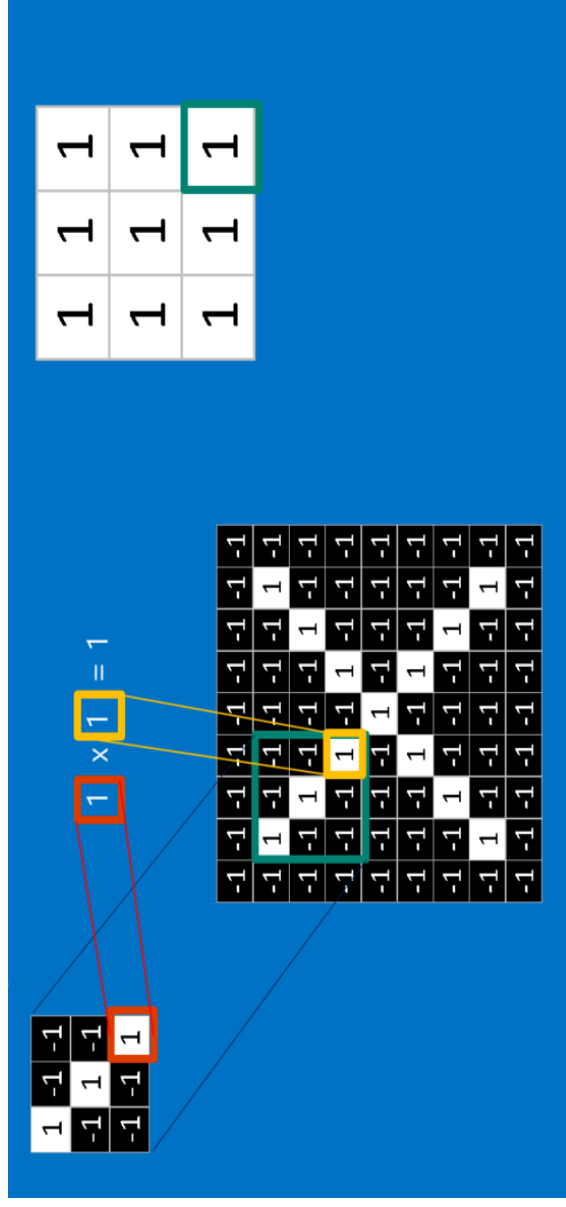


Figure III.1. Convolution sur image.

III.5.3 Pooling

Un autre outil très puissant utilisé par les CNNs s'appelle le Pooling. Le Pooling est une méthode permettant de prendre une large image et d'en réduire la taille tout en préservant les informations les plus importantes qu'elle contient. Les mathématiques derrière la notion de pooling ne sont une nouvelle fois pas très complexe. En effet, il suffit de faire glisser une petite fenêtre pas à pas sur toutes les parties de l'image et de prendre la valeur maximum de cette fenêtre à chaque pas. En pratique, on utilise souvent une fenêtre de 2 ou 3 pixels de côté et une valeur de 2 pixels pour ce qui est de la valeur d'un pas [11].

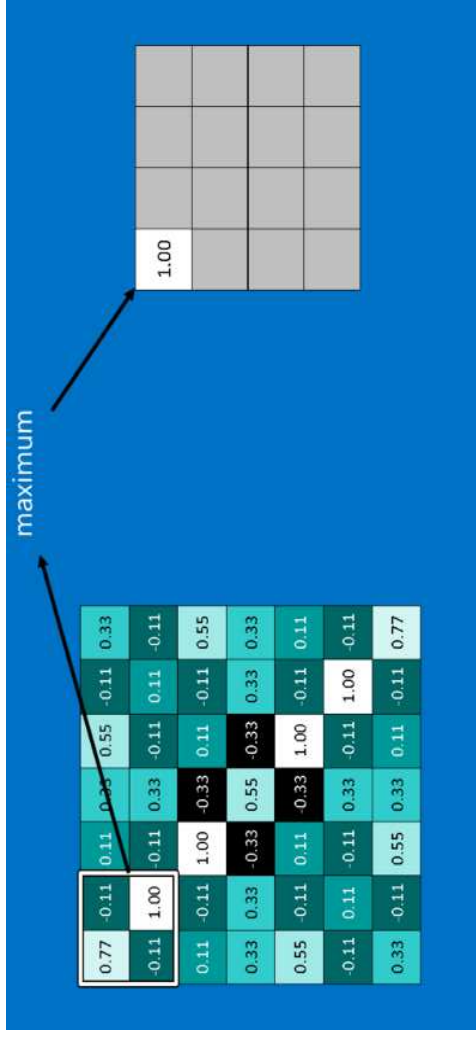


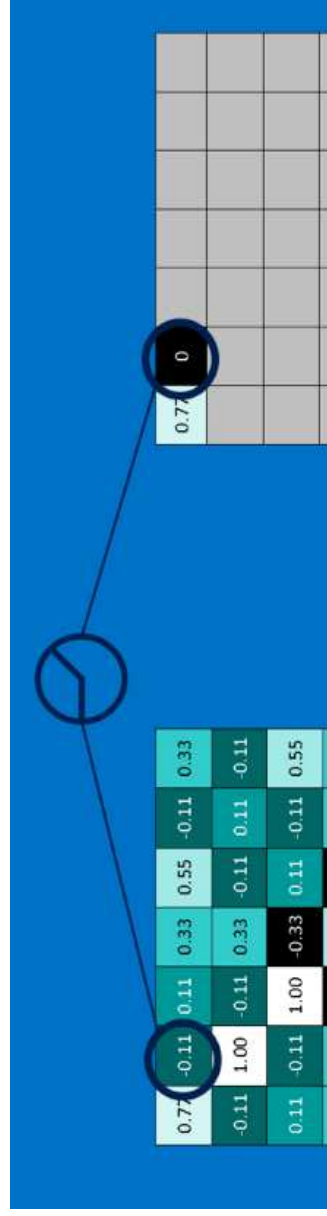
Figure III.2. *Principe de Pooling.*

Après avoir procédé au pooling, l'image n'a plus qu'un quart du nombre de ses pixels de départ.

III.5.4 Unités Rectifié Linéaire (ReLU)

Un élément important dans l'ensemble du processus est l'Unité linéaire rectifiée ou ReLU. Les mathématiques derrière ce concept sont assez simples encore une fois : chaque fois qu'il y a une valeur négative dans un pixel, on la remplace par un 0. Ainsi, on permet au CNN de rester en bonne santé (mathématiquement parlant) en empêchant les valeurs apprises de rester coincer autour de 0 ou d'exploser vers l'infinie.

C'est un outil pas vraiment sexy mais fondamental car sans lequel le CNN ne produirait pas vraiment les résultats qu'on lui connaît [11].



III.5.5 Apprentissage Profond

Les images brutes sont filtrées, rectifiées et mises en commun pour créer un ensemble d'images rétrécies et filtrées par caractéristiques visibles dans chaque image. Celles-ci peuvent ainsi être filtrées et rétrécies encore et encore. A chaque fois, les caractéristiques deviennent plus grandes et plus complexes, et les images deviennent plus compactes. Cela permet aux couches inférieures de représenter des aspects simples de l'image, tels que les bords et points lumineux. Les couches supérieures quant à elles représentent des aspects beaucoup plus complexes de l'image, tels que des formes et des patterns. Ceux-ci ont tendance à être facilement reconnaissables. Par exemple, dans un CNN entraîné à reconnaître des visages, les couches supérieures représentent des motifs et patterns qui font clairement partie d'un visage [11].

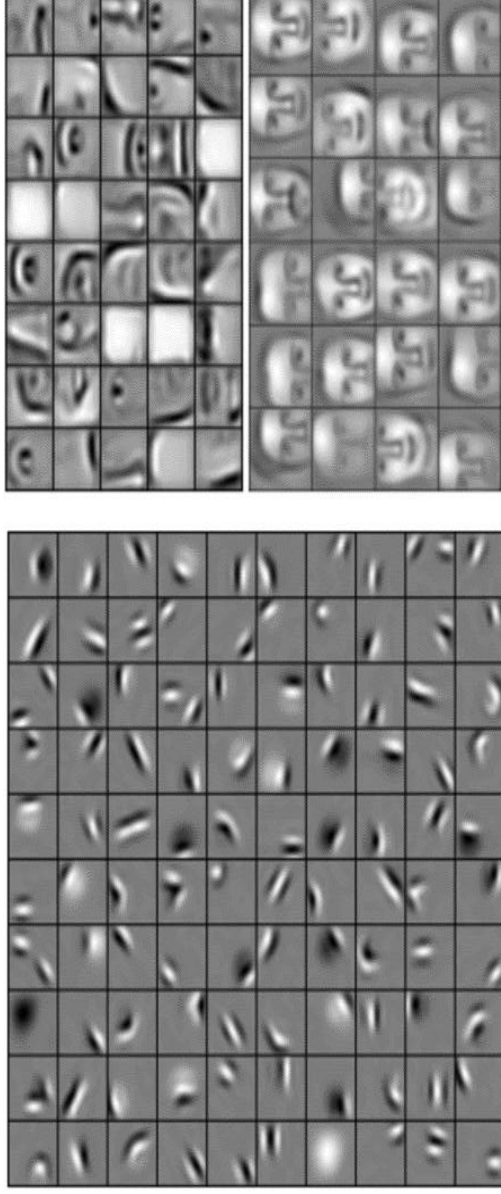


Figure III.4. Apprentissage profond sur visage

III.5.6 Les couches entièrement connectées

Sont les principaux blocs de construction des réseaux de neurones traditionnels. Au lieu de traiter les inputs comme des tableaux de 2 Dimensions, ils sont traités en tant que liste unique

Lorsqu'une nouvelle image est présentée au CNN, elle se répand à travers les couches inférieures jusqu'à atteindre la couche finale entièrement connectée. L'élection a ensuite lieu. Et la solution avec le plus de vote gagne et est déclarée la catégorie de l'image.

Les CNNs ont une autre flèche dans leur carquois. En effet, les couches entièrement connectées prennent les images filtrées de haut niveau et les traduisent en votes. Dans notre exemple, nous devons seulement décider entre deux catégories, X et O [11].

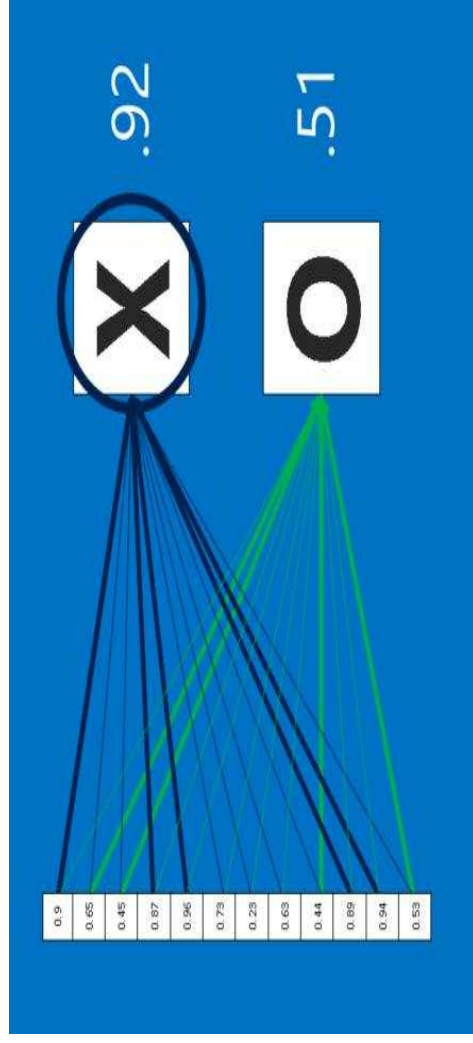


Figure III.5. Les couches entièrement connectées.

III.6 Softmax

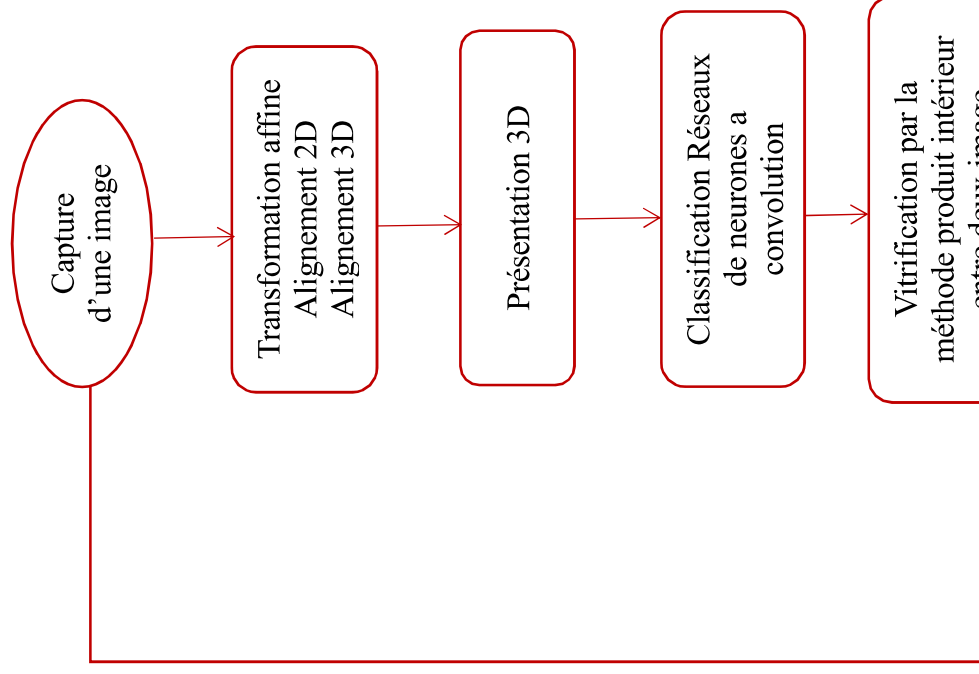
La fonction sigmoïde peut être appliquée facilement, les ReLUs ne disparaissent pas l'effet pendant votre processus d'entraînement. Cependant, quand vous voulez faire face à des problèmes de classification, ils ne peuvent pas aider beaucoup. En termes simples, la fonction sigmoïde ne peut gérer que deux classes, ce qui n'est pas ce que nous attendons [12].



Mathématiquement, la fonction softmax est représentée ci-dessous, où z est un vecteur des entrées de la couche de sortie (si vous avez 10 unités de sortie, il y a 10 éléments dans z). Et encore, j indexe les unités de sortie, donc $j = 1, 2, \dots, K$ [12].

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}}$$

III.7 Organigramme de Système



III.8 Conclusion

Dans ce chapitre, nous avons donné une importance aux méthode Deep Face puisqu'elles seront notre sujet de travail, ensuite nous avons détaillé Transformations affines et nous avons présenté l'utilité de de classifieur softmax dans le traitement des images et nous avons terminé avec la présentation de Réseaux Neurones à Convolution (CNN) comme étant une méthode adoptée dans la phase de comparaison incluse dans le système de reconnaissance.

IV. Réalisation

IV.1 Introduction

Ce chapitre est consacré à la réalisation du système, dans la première partie, on va présenter le matériel et le logiciel nécessaire au fonctionnement du système d'identification a profond, puis donner une définition sur les bibliothèques utilisées et les Framework et l'API.

Tandis que la seconde partie c'est l'implémentation de ce système où je vais expliquer les différents résultats obtenus ainsi que les diverses discussions.

IV.2 Environnement matériel et logiciel

Le travail s'effectue sur micro-ordinateur portable.

IV.2.1 Les ressources physiques suivant

- Processeur Intel® Core™ i3-3120M CPU @ 2.50GHz × 4
- Une mémoire vive d'une capacité de 4 Go
- Une carte graphique Gallium 0.4 on AMD TURKS (DRM 2.43.0 / 4.4.0-127-generic, LLVM 3.8.0)

IV.2.2 Pour ce qui est côté logiciel :

- Système d'exploitation : Ubuntu 16.04 TLS de type 64 bits
- Langage de programmation : Python.

IV.3 Langage de programmation

Python est un langage puissant, à la fois facile à apprendre et riche en possibilités. Dès l'instant où vous l'installez sur votre ordinateur, vous disposez de nombreuses fonctionnalités

- Des programmes complets, comme des jeux, des suites bureautiques, des logiciels multimédias, des clients de messagerie...
- Des projets très complexes, comme des progiciels (ensemble de plusieurs logiciels pouvant fonctionner ensemble, principalement utilisés dans le monde professionnel).

Python est un langage de programmation interprété, c'est-à-dire que les instructions que vous lui envoyez sont « transcrites » en langage machine au fur et à mesure de leur lecture. D'autres langages (comme le C / C++) sont appelés « langages compilés » car, avant de pouvoir les exécuter, un logiciel spécialisé se charge de transformer le code du programme en langage machine. On appelle cette étape la « compilation ». À chaque modification du code, il faut rappeler une étape de compilation [17].

Les avantages d'un langage interprété sont la simplicité (on ne passe pas par une étape de compilation avant d'exécuter son programme) et la portabilité (un langage tel que Python est censé fonctionner aussi bien sous Windows que sous Linux ou Mac OS, et on ne devrait avoir à effectuer aucun changement dans le code pour le passer d'un système à l'autre) [17].

IV.4 Les bibliothèques utilisées

IV.4.1 TensorFlow

Nous en entendons beaucoup parler ces derniers temps, et pour cause, TensorFlow est devenu en un temps record l'un des Framework de référence pour le Deep Learning, utilisé aussi bien dans la recherche qu'en entreprise pour des applications en production. Au-delà du type présente autour de ce Framework et des projets qui émergent grâce à ce dernier, il reste un gap non négligeable à atteindre afin de l'utiliser pleinement et efficacement [14].

TensorFlow est un Framework de programmation pour le calcul numérique qui a été

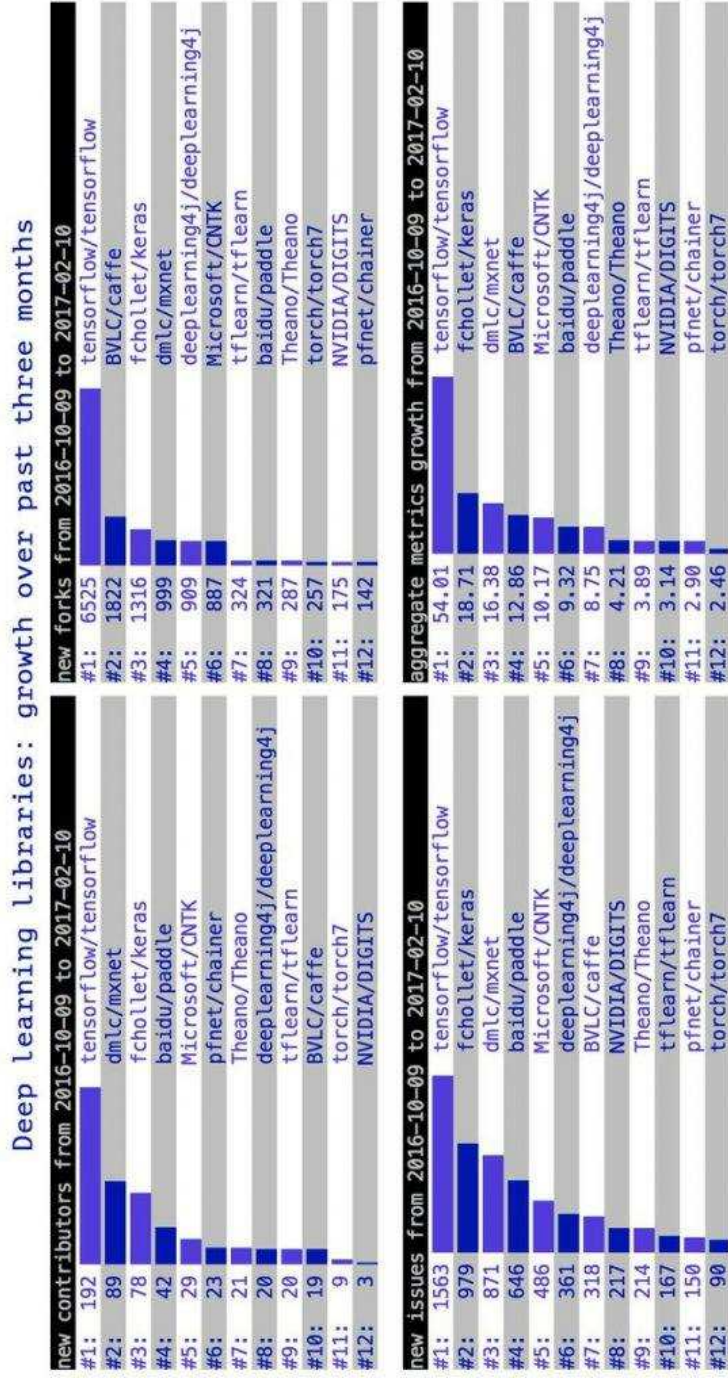


Figure IV.1. La bibliothèque plus utilisée dans trois mois.

Les raisons de cette popularité fracassante, elles sont nombreuses :

Multi-plateformes (Linux, Mac OS, et même Android et iOS)

- APIs en Python, C++, Java et Go (l'API Python est plus complète cependant, c'est sur celle-ci que nous allons travailler).
- Temps de compilation très courts dû au backend en C/C++.
- Supporte les calculs sur CPU, GPU et même le calcul distribué sur cluster.
- Une documentation extrêmement bien fournie avec de nombreux exemples et tutoriels.
- Last but not least : Le fait que le Framework vienne de Google et que ce dernier ait an-

est destinée à être utile dans la recherche et les projets commerciaux du monde réel et a été soigneusement conçue pour faciliter l'intégration dans l'application C ++ d'un utilisateur [16].

IV.4.3 Comment ça marche TensorFlow

La particularité de TensorFlow est qu'il représente les calculs sous la forme d'un graphe d'exécution : chaque nœud représente une Opération à réaliser, et chaque lien représente un Tensor. Une Opération peut aller d'une simple addition à une fonction complexe de différenciation matricielle [14].

Chaque Opération prend en entrée zéro, un ou plusieurs Tensor, effectue un calcul, et retourne zéro, un ou plusieurs Tensor. Un exemple typique de Tensor est un batch d'images. Un batch d'images est représenté par un Tensor à 4 dimensions : taille du batch (nombre d'images dans le batch), hauteur, largeur et nombre de canaux de représentation (3 pour une image en couleurs représentée en RGB) [14].

La création du graphe est automatiquement gérée par TensorFlow une fois les Tensor et Opération implémentés et instanciés. Cela permet une optimisation et parallélisation du code et de l'exécution lors du lancement [14].

TensorFlow possède de plus un support très vaste pour la création d'opérations spécifiques au Deep Learning, et il devient donc facile de construire un réseau de neurones et d'utiliser les opérations mathématiques couramment associées pour l'entraîner avec les bons optimiseurs [14].

IV.4.4 Le modèle inception-v3

Le modèle Inception v3 est un réseau de neurones convolutionnels profonds, qui a été pré-entraîné pour le défi de reconnaissance visuelle Large, et il peut différencier entre 1000 classes différentes [15].

IV.5 Système iKnow you

IV.5.1 Détection

Les histogrammes de gradients orientés (HOGs) sont la concaténation d'histogrammes obtenus en divisant l'image en petites régions connectées appelées cellules. Ainsi, l'apparence et la forme de l'objet local dans une image peuvent être décrites par la distribution de gradients d'intensité ou de directions de bord. Les descripteurs HOG sont particulièrement adaptés à la détection humaine car il a été démontré qu'ils surpassent d'autres ensembles de fonctionnalités dans cette tâche [14].

L'avantage de la fonctionnalité HOG est qu'elle est basée sur histogramme du gradient orienté. Il ne peut pas seulement décrire Soit la caractéristique des contours du visage, mais aussi ne pas être sensible à la lumière et petit décalage. Obtenir les traits du visage humain en combinant les caractéristiques de tous les blocs en ligne. Les modèles de visualisation de visage aident à développer une détection de visage algorithme basé sur HOG descripteurs exemples de détection est montré à la (figure 3) et les spécifications conçues sont :
Fréquence d'images = 25.0000, Gain = 29, Gamma = 1, Teinte = 0, Saturation = 13, taille du bloc pour le visage non maximal = 0,3, seuil = 0,6, Box cache size = 100000, temps de détection= 8.6705secs.

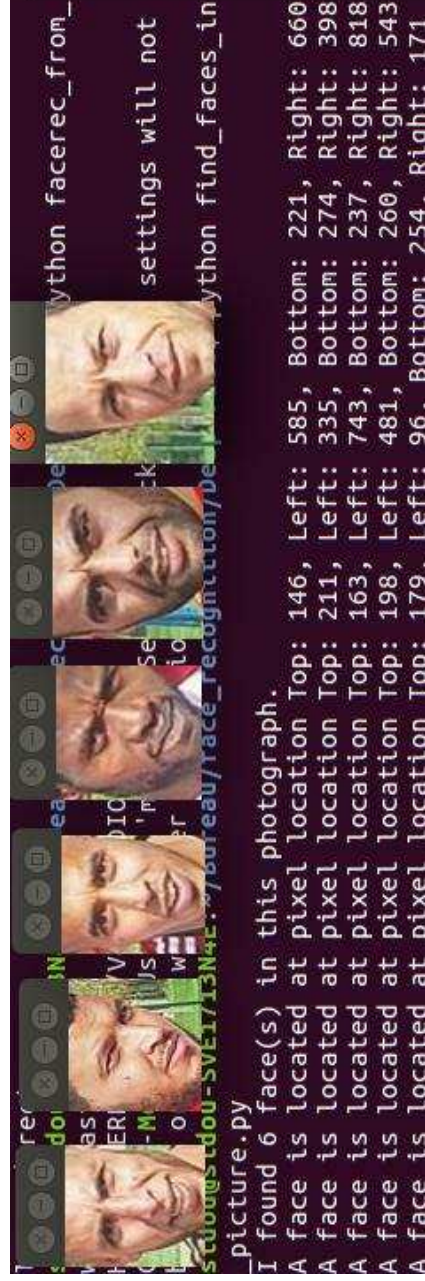




Figure IV.4. Détecter les repères faciaux 2D en images

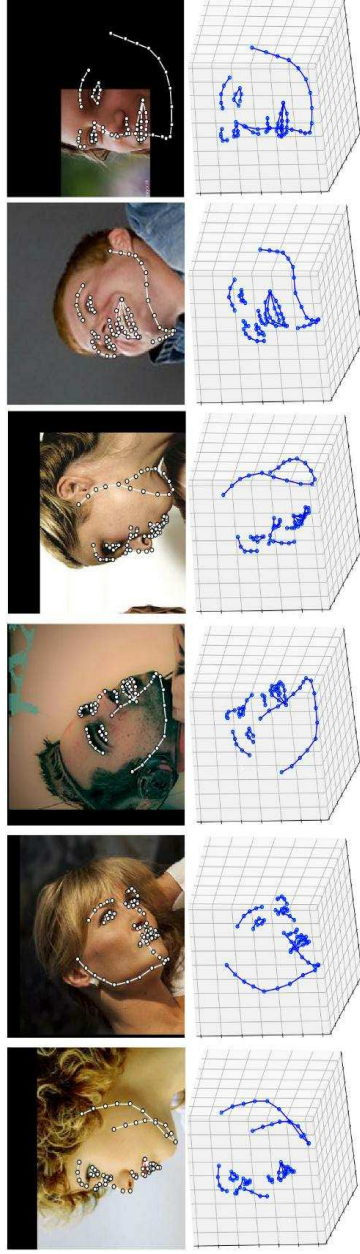


Figure IV.5. Détecter les repères faciaux 3D en images

IV.5.3 Classification

IV.5.3.1 Pour Image :

IV.5.3.1.1 Les données

La base donnée utilisé est simple pour accélérer l'exécution mais il y a un manque de matériels.

Quand nous classons l'image du visage de quelqu'un, nous devons ajouter un répertoire 'training_dataset' et remplir les sous-dossiers, nommés par le nom et prénom de personne d'après le label de la classe. Ces dossiers contiendront des ensembles de données (images) pour

```

2018-06-02 20:15:22.018857: Step 0: Train accuracy = 40.0%
2018-06-02 20:15:22.019044: Step 0: Cross entropy = 2.230768
2018-06-02 20:15:22.560152: Step 0: Validation accuracy = 27.0% (N=100)
2018-06-02 20:15:23.702692: Step 10: Train accuracy = 72.0%
2018-06-02 20:15:23.702873: Step 10: Cross entropy = 1.845677
2018-06-02 20:15:23.812142: Step 10: Validation accuracy = 54.0% (N=100)
2018-06-02 20:15:24.969133: Step 20: Train accuracy = 76.0%
2018-06-02 20:15:24.969267: Step 20: Cross entropy = 1.488271
2018-06-02 20:15:25.064198: Step 20: Validation accuracy = 56.0% (N=100)
2018-06-02 20:15:26.198730: Step 30: Train accuracy = 79.0%
2018-06-02 20:15:26.198903: Step 30: Cross entropy = 1.412381
2018-06-02 20:15:26.316116: Step 30: Validation accuracy = 67.0% (N=100)
2018-06-02 20:15:27.438697: Step 40: Train accuracy = 85.0%
2018-06-02 20:15:27.438800: Step 40: Cross entropy = 1.211712
2018-06-02 20:15:27.532105: Step 40: Validation accuracy = 59.0% (N=100)
2018-06-02 20:15:28.574348: Step 50: Train accuracy = 80.0%
2018-06-02 20:15:28.574543: Step 50: Cross entropy = 1.193379
2018-06-02 20:15:28.694886: Step 50: Validation accuracy = 78.0% (N=100)
2018-06-02 20:15:29.748263: Step 60: Train accuracy = 91.0%
2018-06-02 20:15:29.748374: Step 60: Cross entropy = 1.051684

```

Figure IV.6. Training le modale.

Une fois le processus terminé, il retournera une précision d'entraînement autour de 90% à peu près.

Les étiquettes recyclées, les graphiques et le résumé de l'entraînement seront sauvegardés dans un dossier nommé « tf_files ».

Après avoir recyclé le modèle, il est maintenant temps de tester le modèle sur nos propres images.

Un exemple sur une identification qui est dans la photo :



IV.5.3.2 Pour vidéo :

IV.5.3.2.1 Détection et Alignement

Pour la détection on a choisi la méthode MTCNN : La détection et l'alignement des visages sont basés sur le papier "Détection et alignement des visages en utilisant des réseaux neuronaux convolutionnels en cascade multifonctions Cette méthode de détection des visages a un avantage sur divers états de lumière, variations faciales et variations visuelles du visage. L'encastrement du visage est une représentation vectorielle multidimensionnelle d'un visage qui représente l'identité unique du visage. Facenet a utilisé 128 dimensions et a créé un modèle qui cartographie tout visage humain en générique. Lorsque nous fournissons une image d'entrée au modèle, elle nous donne 128 octets de données vectorielles numériques qui peuvent être générés par comparaison avec la représentation générique du visage mappé du modèle. Ces points d'inclusion sont facilement comparables en mesurant la distance euclidienne [13].

Et cette méthode se base sur trois étapes :

P-net \rightarrow R-net \rightarrow O-net.

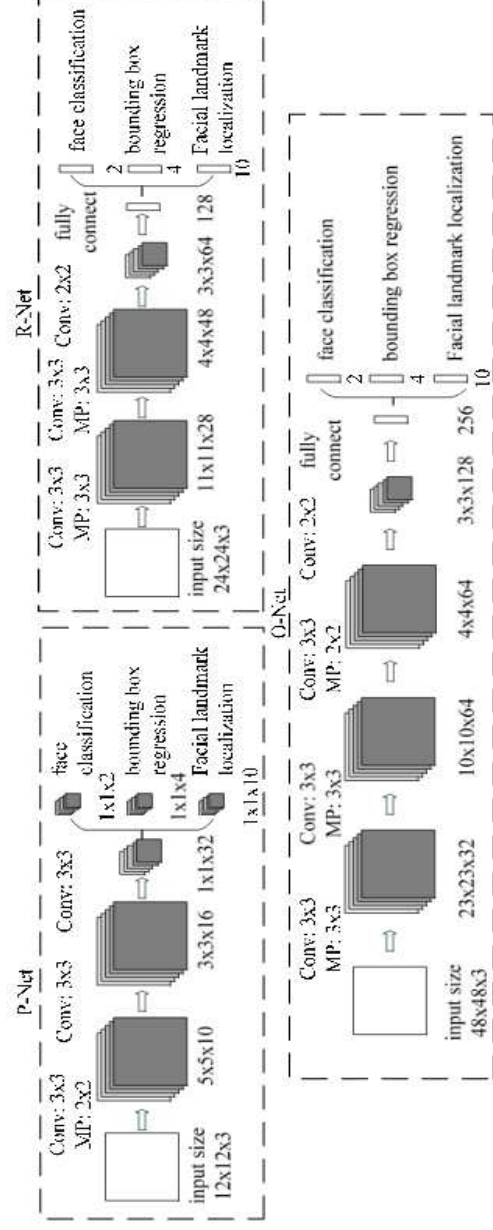


Figure IV.8 Les architectures de P-Net, R-Net et O-Net

- Dans la troisième étape, le réseau de sortie (O-Net) produit la zone de délimitation finale et la position des repères faciaux.

IV.5.3.3 Caractéristique de visage

L'encastrement du visage est une représentation vectorielle multidimensionnelle d'un visage qui représente l'identité unique du visage. Notre méthode a utilisé 128 dimensions et a créé un modèle qui cartographie tout visage humain en générique. Lorsque nous fournissons une image d'entrée au modèle, elle nous donne 128 octets de données vectorielles numériques qui peuvent être générés par comparaison avec la représentation générique du visage mappé du modèle. Ces points d'inclusion sont facilement comparables en mesurant la distance euclidienne.

```
{
  "GorLine": {
    "Right": [
      [-0.0502917654812336, 0.1176426112651825,
        0.01324997004121542, -0.02534906193614006, -0.0564335472881794,
        0.0761787168821335, -0.025162912905216217, -0.05786031857132912,
        -0.051409557461738586, 0.018408669158816338, 0.04539864882826805,
        0.10730986297130585, 0.15238657593727112, -0.19706012308597565,
        0.0146661512337625027, 0.027142584323883057, 0.0032476806081831455,
        0.023905299603939056, -0.004226752556860447, 0.04069270193576813,
        -0.006275825202465057, 0.06542107462882996, -0.04211834445595741,
        0.0439201295375824, 0.08931898325681686, -0.016929449513554573,
        -0.002458695551058054, 0.04718736559152603, -0.05731166526675224,
        -0.08599866764450073, -0.0100289611145854, -0.048850130289793015,
        -0.021939115598797798, -0.023899078369140625, 0.0774878362064362,
        -0.07269284874200821, 0.096326544880867, -0.10853684693574905, 0.03697269782423973,
        0.03837412968277931, 0.052315711975097656, 0.04937761649489403,
        0.01165036205202341, 0.1589156687259674, 0.031571920961141586,
        -0.08270739763975143, 0.01282272022175121, -0.019474158063530922,
        0.0743037685751915, -0.017151178792119026, -0.04729205369949341,
        0.03341216221451759, 0.008544989861547947, 0.04334603250026703,
        0.00630639772862196, 0.05747710168361664, 0.008158858865499496,
        -0.12257859110832214, -0.018593354150652885, 0.014334550127387047,
        0.16841815412044525, -0.07990828901529312, 0.020622311159968376,
        -0.08073481917381287, 0.012534964829683304, -0.0250532161444256,
        -0.05925936624407768, 0.0061850412748754025, -0.12452470511198044,
        -0.10606827586889267, -0.10948717594146729, 0.11249823123216629,
        -0.0525302030146122, -0.09253156930208206, 0.13406182825565338,
        -0.0894979685449677, 0.00469094468280673, 0.1507372260093689,
        0.045905096204366684, 0.046475473791360855, -0.0270614642649889,
        -0.0003507397195789963, -0.11715945601463318, 0.050579845905303955,
        0.04212042689323425, 0.0066687376238405704, -0.027750791981816292,
        -0.09523145854473114, -0.07875782996416092, -0.037278734147548676,
        -0.001691056936979294, -0.05368301644921303, -0.03037908673286438,
        -0.09182731807231903, 0.1376463621854782, -0.037520881742239, 0.02109302021563053,
```


$$\begin{aligned}
 d(\mathbf{p}, \mathbf{q}) &= d(\mathbf{q}, \mathbf{p}) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \dots + (q_n - p_n)^2} \\
 &= \sqrt{\sum_{i=1}^n (q_i - p_i)^2}.
 \end{aligned}$$

Figure IV.10. La formule distance euclidienne.

IV.5.3.4 Les bibliothèques utilisées

- OpenCV : Traitement d'image (VideoCapture, redimensionnement etc.).
- Tensorflow: Le fameux Deep Learning Framework de Google.

IV.5.4 Capture sur l'interface

IV.5.4.1 Interface générale

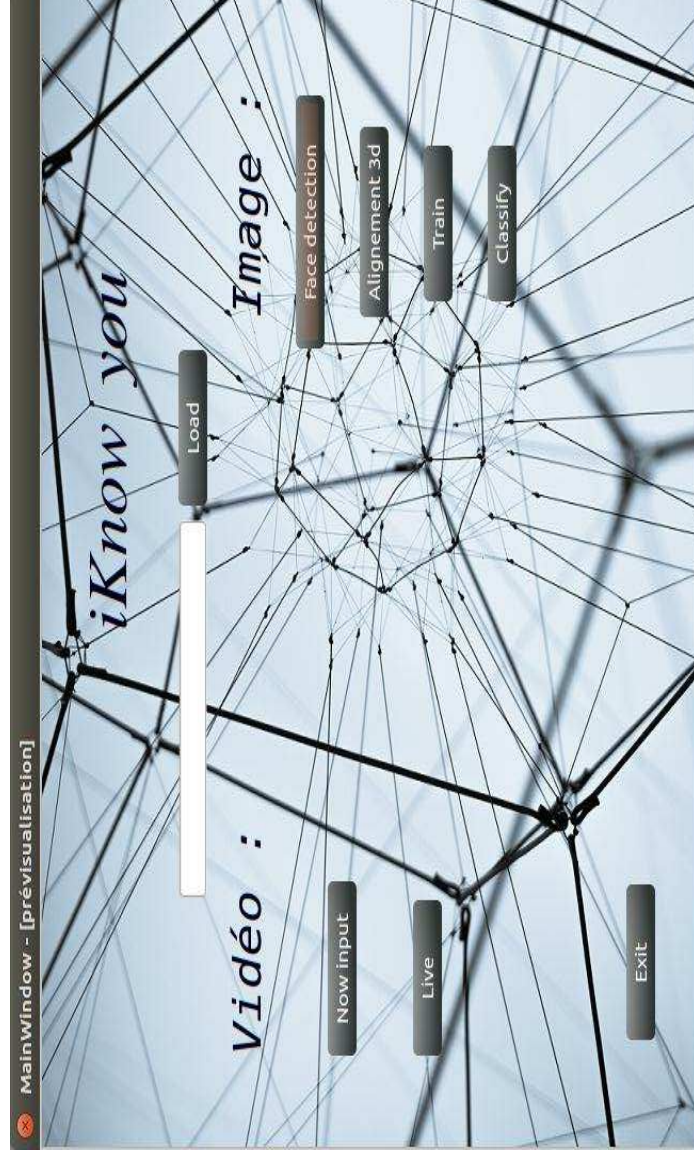


Figure IV.11. Interface générale.

IV.5.4.2 Insertion d'un identifiant

On clique sur Bouton « Now input », sinon on entre dans le terminal « ``python3 main.py --mode"input" `` » pour ajouter un nouvel utilisateur. Commencez à tourner à gauche, à droite, en haut, en bas après avoir entré le nouveau nom. Tournez lentement pour éviter les images floues.

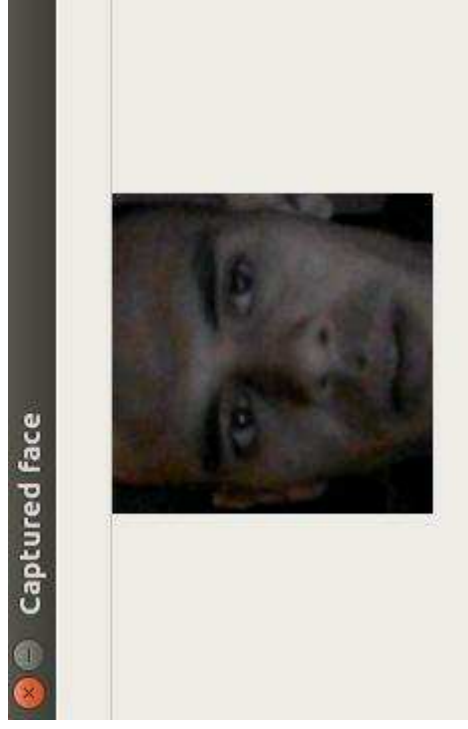
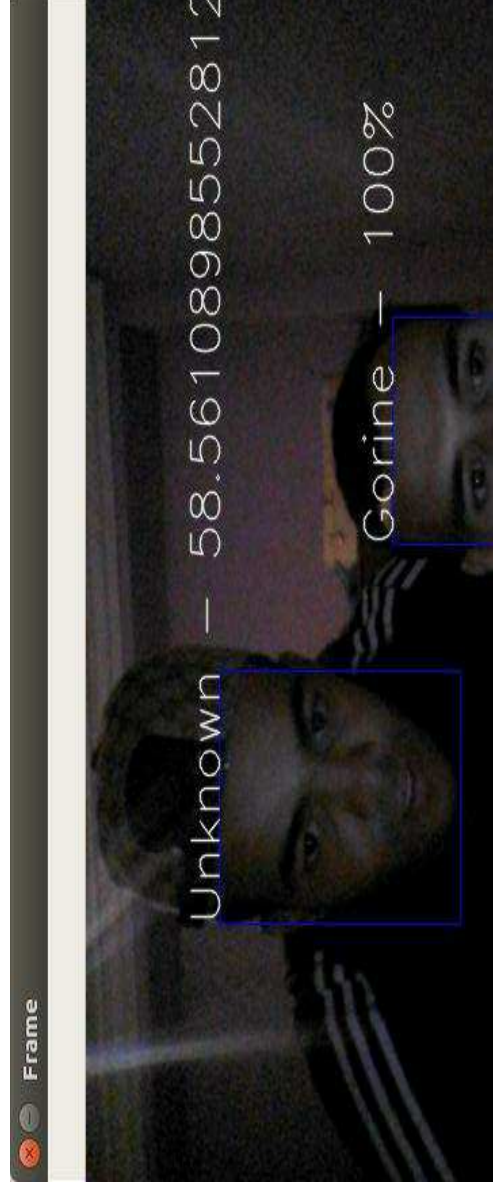


Figure IV.12. Mode entrer une personne dans la base.



IV.5.4.3 Identifier en directe à partir de caméra

On clique sur le Botton « live », sinon on entre dans le terminal « `python3 main.py` » pour lancer le programme.

```
( ) Timings at 240p:  
- Face locations: 0.0879s (11.37 fps)  
- Face landmarks: 0.0034s (291.73 fps)  
- Encode face (inc. landmarks): 0.0285s (35.08 fps)  
- End-to-end: 0.1169s (8.55 fps)  
( ) Timings at 480p:  
- Face locations: 0.3377s (2.96 fps)  
- Face landmarks: 0.0036s (279.81 fps)  
- Encode face (inc. landmarks): 0.0287s (34.86 fps)  
- End-to-end: 0.3672s (2.72 fps)  
( ) Timings at 720p:  
- Face locations: 0.7677s (1.30 fps)  
- Face landmarks: 0.0036s (279.02 fps)  
- Encode face (inc. landmarks): 0.0286s (34.92 fps)  
- End-to-end: 0.7967s (1.26 fps)  
( ) Timings at 1080p:  
- Face locations: 1.7170s (0.58 fps)  
- Face landmarks: 0.0037s (273.52 fps)  
- Encode face (inc. landmarks): 0.0289s (34.55 fps)  
- End-to-end: 1.7574s (0.57 fps)  
( )
```

Figure IV.14. Résultat de la précession par apport la qualité de vidéo.

IV.5.5 Visualisation de donnée

Le temps de réponse de système « I-Know you » fait l'identification très rapidement par ce que notre base de données n'est pas volumineuse, et plus on a adopté de Fremwork et des bibliothèques propres et ça il est minimisé la complexité du programme. Mais même dans le cas où la base de données volumineuse notre système offre la possibilité d'identifier une personne très rapidement.

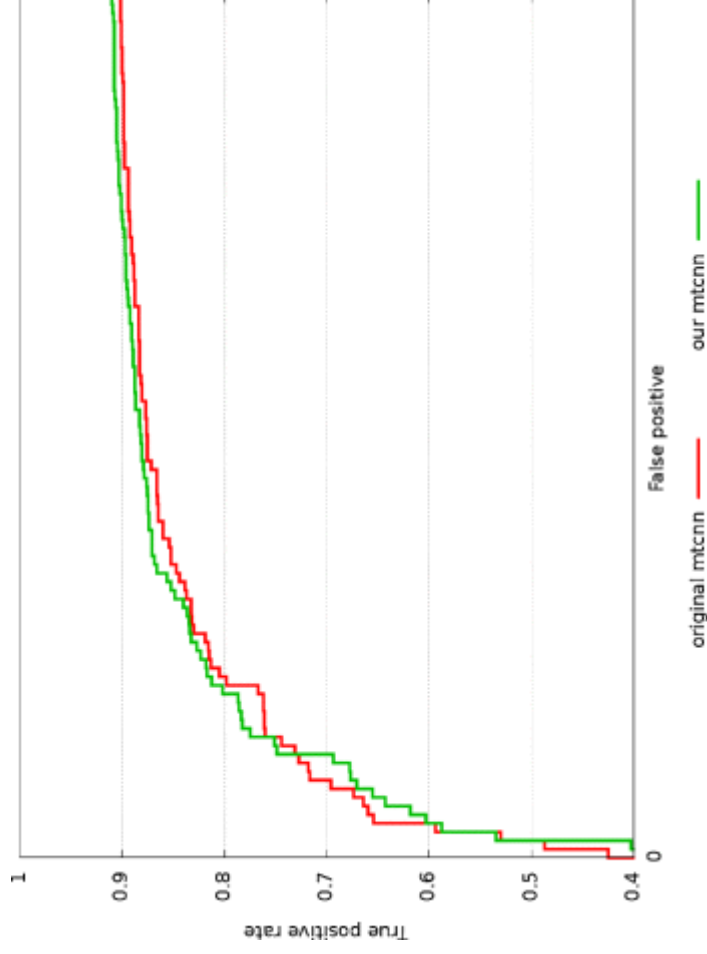


Figure IV.15. Résultat de détection en LFW par MTCNN.

IV.5.5.2 Apprentissage

Une fois les goulots d'étranglement terminés, la formation proprement dite de la couche supérieure du réseau commence. Vous verrez une série de résultats d'étape, chacun montrant la précision de l'entraînement, la précision de la validation et l'entropie croisée. La précision de l'entraînement montre quel pourcentage des images utilisées dans le lot d'entraînement actuel ont été étiquetées avec la bonne classe. La précision de la validation est la précision sur un groupe d'images sélectionné au hasard à partir d'un ensemble différent. La principale différence est que la précision de l'entraînement est basée sur des images que le réseau a pu apprendre afin que le réseau puisse s'adapter au bruit dans les données d'entraînement. Une véritable mesure de la performance du réseau consiste à mesurer sa performance sur un ensemble de données qui ne figure pas dans les données d'apprentissage - ceci est mesuré par la précision de la validation.

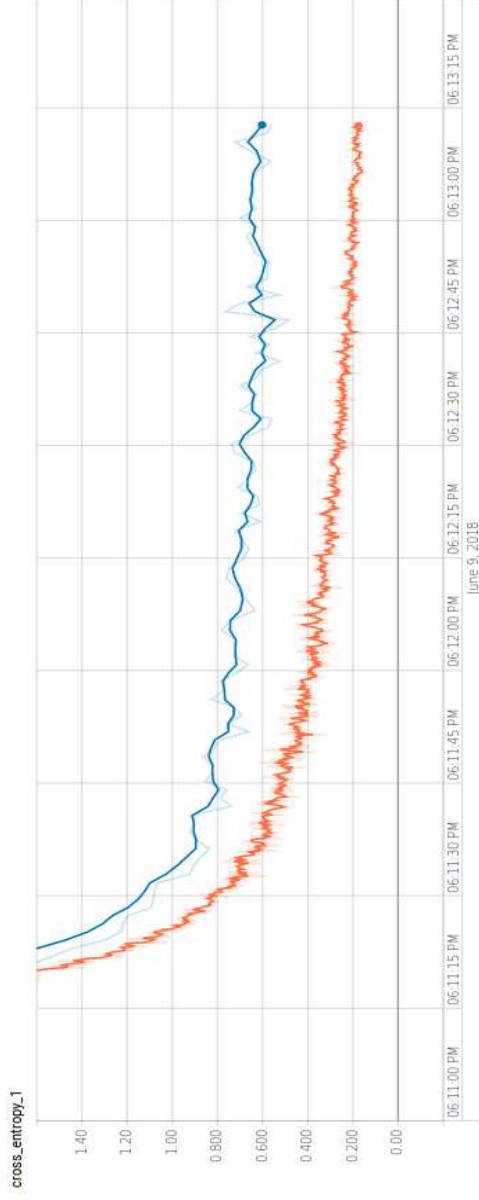
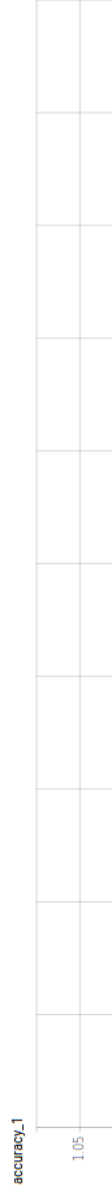


Figure IV.16. Résultat de l'entropie croisée et la validation.

Par défaut, ce script exécutera 4 000 étapes d'entraînement. Chaque étape choisit au hasard dix images de l'ensemble d'apprentissage, trouve leurs goulots d'étranglement dans le cache et les introduit dans la couche finale pour obtenir des prédictions. Ces prédictions sont ensuite comparées aux étiquettes réelles pour mettre à jour les poids de la couche finale à travers le processus de rétropropagation. Au fur et à mesure que le processus se poursuit, vous devriez voir la précision rapportée s'améliorer, et une fois toutes les étapes terminées, une évaluation finale de la précision du test est effectuée sur un ensemble d'images séparées des images de formation et de validation. Cette évaluation de test est la meilleure estimation de la façon dont le modèle entraîné effectuera la tâche de classification. Vous devriez voir une valeur d'exactitude comprise entre 90% et 95%, bien que la valeur exacte variera d'une course à l'autre car il y a un caractère aléatoire dans le processus d'entraînement. Ce nombre est basé sur le pourcentage d'images dans l'ensemble de test qui reçoivent l'étiquette correcte une fois que le modèle a reçu une formation complète[14].



IV.6 Conclusion

Dans ce chapitre, on a décrit les étapes de réalisation du système d'identification profond « iKnow-you », l'environnement matériel et logiciel, ainsi que les détails de fonction utilisée et les bibliothèques adoptées de réalisation de notre système et l'interface développée. En vue de ce qui a été fait dans la réalisation, on peut dire que :

- **IKnow you** est très interactif, en effet en plus d'identification avec deux méthodes de détection et il permet aussi d'ajouter de nouvelle personne à la base de données.
- **IKnow you** est performant en raison des bons taux d'identification surtout si la vidéo est de bonne qualité.
- Son interface est simple que toute personne ayant peu d'identifier sur la caméra de surveillance ou dans les portes de sécurité.
- L'identification est très rapide, vu que notre base de données n'est pas volumineuse.

Conclusion Générale

Le but pratique de la reconnaissance faciale profond est d'avoir un système performant et applicable dans la réalité, malgré tous les problèmes existants qui perturbent la qualité de ces systèmes.

Dans ce rapport, nous avons présenté les technologies utilisées dans les systèmes pour l'identification de personnes sur Facebook, Nous avons aussi donné un aperçu sur les techniques de mesure de leurs performances. Cette étude nous a permis de constater que la reconnaissance de visages suscite de plus en plus l'intérêt de la communauté scientifique, car elle présente plusieurs challenges et verrous technologiques. Nous avons présenté les techniques de la reconnaissance faciale en utilisant plusieurs approches parmi elles nous avons choisi l'approche d'apprentissage profond.

L'algorithme de Réseaux de Neurones à Convolution est une méthode très compliquée, il offre une bonne performance de classification des personnes connues. Au plus il est adopté dans plus Réseaux sociaux et Google, les problèmes de pose, d'éclairage, et d'identification ; dans des environnements extérieurs ; restent des challenges qui susciteront les efforts des chercheurs.

Les perspectives de ce travail sont nombreuses : dans un premier temps, on souhaite développer un module qui fait présenter le visage en forme 3D et essayer de traiter tous les points de visage et faire une classification sur le modale 3D mais puis que CNN est complexe et prend du temps pour faire la classification car il prend que les point des yeux et le nez. Il serait intéressant de définir un domaine variable portant sur la variété de condition d'éclairage,

Bibliographies

- [1] O. Barkan, J. Weill, L. Wolf, and H. Aronowitz. Fast high dimensional vector multiplication face recognition. In ICCV, 2013.
- [2] Cécile Fiche. Repousser les limites de l'identification faciale en contexte de vidéo-surveillance. Université de Grenoble, mémoire pour l'obtention de grade de maître sciences, Institut de la francophonie pour l'Informatique, 2012.
- [3] W.Hizem, « Capture Intelligent pour la Reconnaissance de visages » Thèse de doctorat, Option : Informatique/Electronique, Institut National des Télécommunication, Université Pierre et Marie Curie, Paris, 2009 ;
- [4] M. Turk and Pentland, « Eigenfaces for Recognition », in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Vision and Modeling Group, the Media Laboratory, Massachusetts Institut of Technology, Cambridge, USA, June 1991, pp.586-591.
- [5] R. Al Seddik et A. Shahin, « Reconnaissance du Mouvement du Visage à partir des Données Vidéo », Mémoire d'Ingéniorat en Électricité et Électronique Option Informatique Industrielle & Contrôle, Université Libanaise Faculté de Genie, Soutenu en Juillet 2013.
- [6] A .Hamadi S.Limam M.Ould Mamar, Développement d'un système de reconnaissance de visage, Mémoire de fin d'étude pour l'obtention du diplôme d'Ingénieur d'Etat en informatique ,Mostaganem 2007/2008.

[9] G. B. Huang, H. Lee, and E. Learned-Miller. Learning hierarchical representations for face verification with convolutional deep belief networks. In Conference on Computer Vision and Pattern Recognition, 2012.

[10] Transformation affine, <http://www.flegrand.fr/scidoc/docimg/graphie/geometrie/affine.html> , consulté le 12/02/2018.

[11] Brandon Rohrer, 'article de (Senior Data Scientist à Facebook), "How Convolutional neural network work ", August 18, 2016.

[12] Deep learning, <https://github.com/Kulbear/deep-learning-nano-foundation/wiki/ReLU-and-Softmax-Activation-Functions> , consulté le 3/02/2018 .

[13] Joint Face Detection and Alignment using Multi-task Cascaded Convolutional Networks ,Kaipeng Zhang, Zhifeng Li, Yu Qiao, Multimedia Research Center, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences ,Multimedia Laboratory, Department of Information Engineering, The Chinese University of Hong Kong ,IEEE Signal Processing Letters (SPL), vol. 23, no. 10, pp. 1499-1503, 2016.

[14] TENSORFLOW & DEEP LEARNING <https://blog.xebia.fr/2017/03/01/tensorflow-deep-learning-episode-1-introduction/>

[15] Deep Learning with Tensorflow: Part 2—Image classification <https://towardsdatascience.com/deep-learning-with-tensorflow-part-2-image-classification-58fedffa7b84>

[16] Davis E. King, Dlib a Machine Learning Toolkit, journal of Machine Learning Research 10 (2009) 1755-1758. Northron Grumman ES. ATR and Image Exploitation