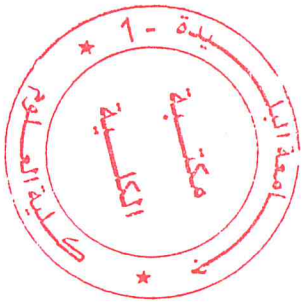
REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE SAAD DAHLEB BLIDA I



Faculté des Sciences

Département d'Informatique

Mémoire présenté par :

CHAIB EDDOUR Nesrine ABDELBAKI Abir

En vue d'obtenir le diplôme de Master

Domaine : Sécurité des Systèmes d'Informations

Thème :
Alignement de visage pour la reconnaissance faciale

Filière : Informatique

Spécialité : Informatique

Option : Sécurité des Systèmes d'Informations

Lieu de stage : Centre de Développement des Technologies Avancées CDTA

Président	Mr OULD AISSA
Examinateur	Mme Yakhlef
Promoteur	Mr KAMECHE
Encadreur	Mme AIT SADI Karima

Année universitaire 2017-2018

Remerciements

C'est avec un immense plaisir que nous réservons cette page, en signe de gratitude et de reconnaissance à tous ceux qui nous ont aidés dans l'accomplissement de ce travail

Tout d'abord, nous souhaitons exprimer nos sincères remerciements à ALLAH LE TOUT PUISSANT qui nous a donné la force, le courage, la volonté et la patience durant toutes ces années d'études et qui a guidé nos pas pour aller jusqu'au bout de ce travail.

Nous tenons avant tout à exprimer nos sincères et chaleureux remerciements à Mme Ait Saadi du CDTA notre Encadreuse, et Mr A. Kameche de l'Université Blida 1 Saad Dahleb, notre Promoteur, qui nous ont permis tous les deux de poursuivre notre projet. Leur présence, leur aide et leur soutien multiformes nous ont été précieux et d'un apport inestimable.

Nous remercions aussi chaleureusement les Membres de cet Honorable Jury pour l'honneur qu'ils nous ont fait par leur présence ici aujourd'hui en acceptant d'évaluer notre travail.

Nous remercions également tous les Enseignants du Département d'Informatique de l'Université Blida1 Saad Dahleb qui durant nos longues années d'études ont toujours su nous inculquer avec efficacité toutes les connaissances que nous avons capitalisées durant notre cursus universitaire.

Enfin nous ne pouvons pas clore cette toute dernière partie de notre travail sans remercier tous les membre de nos familles, nos amis, et plus particulièrement nos parents pour les coups de pousse qu'ils nous ont depuis toujours donné et surtout pour tout le soutien qu'ils nous ont apporté avec beaucoup de patience et de tendresse.

Résumé

La détection faciale est actuellement un domaine en plein essor. Elle rentre petit à petit dans nos vies au travers de nos téléphones mobiles ou de nos ordinateurs portables. Malgré l'amélioration du taux de détection elle reste actuellement l'objet de nombreuses études et de nombreux travaux d'approfondissement. L'objectif de notre projet sera de mettre en œuvre un système de détection et alignement de visage pour pouvoir ultérieurement continuer la reconnaissance faciale. Nous effectuerons cette détection et alignement facial sur des photos ainsi que des vidéos en temps réel.

Nous vous présenterons la technique permettant la localisation des repères faciaux. Cette technique est basée sur les réseaux de neurones Convolutionnels (CNN) qui sont très connus. L'objectif du mécanisme de détection et d'alignement du visage que nous vous présenterons est d'effectuer les opérations suivantes :

- La localisation automatique du visage dans une image.
- La localisation des points de repères faciaux.
- La localisation automatique du visage et des points de repères faciaux en temps réel dans toutes les positions

Pour cela nous avons utilisé les bibliothèques OpenCV, Dlib, et caffe
Mots clés : détection faciale, alignement de visage, reconnaissance faciale, réseaux de neurones Convolutionnels (CNN).

Abstract

Facial detection is currently a growing field. It is gradually returning to our lives through our mobile phones or our laptops. Despite the improvement of the detection rate, it is currently the subject of numerous studies and numerous in-depth studies. The goal of our project will be to implement a face detection and alignment system to be able to continue facial recognition later on. We will perform this detection and facial alignment on photos as well as real-time videos.
We will present you the technique allowing the localization of the facial references. This technique is based on convolutional neural networks (CNN) which are well known. The purpose of the Face Detection and Alignment mechanism that we will be presenting to you is to do the following:

- The automatic location of the face in an image.
- The location of facial landmarks.
- Automatic location of face and facial landmarks in real time in all positions

For this we used the libraries OpenCV, Dlib, and caffe
Key words: facial detection, facial alignment, facial recognition, convective neural networks (CNN).

ملخص

يعد اكتشاف الوجه حاليًا مجالًا متناميًا. إنه يعود تدريجيا إلى حياتنا من خلال هواتفنا المحمولة أو أجهزة الكمبيوتر المحمولة الخاصة بنا. على الرغم من تحسن معدل الكشف ، فهو حاليا موضوع العديد من الدراسات والعديد من الدراسات المتعمقة.
سيكون الهدف من مشروعنا هو تنفيذ نظام للكشف عن الوجه والمحاذاة ليتمكن من مواصلة التعرف على الوجه في وقت لاحق. سنقوم بإجراء هذا الكشف ومحاذاة الوجه على الصور بالإضافة إلى مقاطع الفيديو في الوقت الفعلي.
سنقدم لك التقنية التي تسمح بتوطين مراجع الوجه. تعتمد هذه التقنية على الشبكات العصبية التحويلية (CNN) المعروفة جيداً. الغرض من آلية اكتشاف الوجه والمواءمة التي سنقدمها لك هو القيام بما يلي:

- الموقع التلقائي للوجه في الصورة.
- موقع معالم الوجه.
- الموقع التلقائي لمعالم الوجه والوجه في الوقت الحقيقي في جميع المواقف

لهذا استخدمنا مكتبات OpenCV، Dlib ، وكافيه
الكلمات المفتاحية: كشف الوجه ، محاذاة الوجه ، التعرف على الوجه ، الشبكات العصبية الحملية (CNN)

Introduction générale

Notre projet de fin d'études, intitulé « Alignement de visage pour la reconnaissance faciale » est proposé par le centre d'accueil du « Centre de Développement des Technologies Avancées » CDTA, qui est un établissement public à caractère scientifique et technologique.

Au cours des dernières années, des progrès considérables ont été réalisés dans le domaine de la détection de visage à partir d'image.

Ce progrès est dû aux nombreux travaux dans ce domaine et à la disponibilité des bases de donnée contenant un grand nombre d'image qui ont permis aux chercheurs de progresser de manière crédible dans l'exécution de leurs approches dans ce domaine, avec la possibilité de les comparer à d'autres approches qu'ils utilisent les mêmes bases.

Dans la fin des années 1980 Yan le Cun a développé un type de réseau particulier qui s'appelle le réseau de neurone convolutionnel, ces réseaux sont une forme particulière de réseau neuronal multicouche dont l'architecture des connexions est inspirée de celle du cortex visuel des hêtres humain.

En 2012 plusieurs événements sont soudainement survenus.

Tout d'abord, les GPU (GraphicalProcessing Unit) capables de plus de mille milliards d'opérations par seconde sont devenus disponibles pour un prix moins cher.

Ces puissants processeurs spécialisés, se sont avérés être très performants pour les calculs des réseaux neuronaux.

Deuxièmement, plusieurs records en reconnaissance de visage dans des images ont été battus par des réseaux de neurones convolutionnels.

Dans notre projet on va utiliser les réseaux de neurones convolutionnels pour la localisation des points de repères faciaux dans une image.

Pour ce faire, nous avons structuré notre mémoire en cinq chapitres :

- Dans le premier chapitre nous présenterons les méthodes classiques de détection de visage dans une image.
- Le deuxième chapitre sera consacré à la présentation des différentes méthodes de localisation des points de repères faciaux dans une image.
- Le troisième chapitre sera consacré à la description détaillée des réseaux de neurones Convolutionnels (CNN), ainsi que les différentes architectures des réseaux de neurones Convolutionnels.
- Dans le quatrième chapitre, nous exposerons la partie expérimentale de notre travail.
- Dans le cinquième chapitre nous exposerons les résultats obtenus et la comparaison entre les méthodes utilisés.

Table des matières

Liste des figures

Liste des Tableaux

Chapitre I

Méthodes de détection faciale

La détection de visage a été très largement abordée par la communauté du traitement d'images et de la vision par ordinateur. Même si des progrès importants ont été réalisés durant la dernière décennie, des réponses algorithmiques génériques fiables n'ont toujours pas été dégagées. Ainsi, malgré la diminution du temps de calcul des processeurs et l'efficacité des algorithmes de vision, le suivi robuste du visage demeure à explorer. Ce chapitre est scindé en deux parties, Dans la première partie nous donnerons quelques concepts et les applications de la détection du visage et dans la deuxième partie nous passerons en revue quelques méthodes classiques de détection de visage.

I.1 Introduction

La définition de ce qu'est un visage est un problème qui se pose depuis longtemps et qui n'a toujours pas été résolu. Il n'existe pas de critère qui permette de certifier que quelque chose est un visage ou ne l'est pas. Le choix a toujours été subjectif. Bien qu'identifier une personne à partir de son visage soit une tâche aisée pour un être humain, elle reste une tâche difficile pour les ordinateurs. La difficulté associée à la détection de visage par la machine peut être attribuée à de nombreux facteurs : les variations d'échelle, l'emplacement, l'orientation, l'expression du visage, les conditions d'éclairage, les occlusions, etc. Malgré des progrès très conséquents dans le domaine du traitement d'image et dans celui de la diminution du temps de calcul des processeurs, des algorithmes génériques de détection n'ont toujours pas été dégagés

I.2 Pourquoi la détection de visages ?

La détection de visages est le fait de trouver les coordonnées spatiales délimitant un visage dans une image ou une vidéo. En termes simples, cela revient à trouver les carrés qui délimitent le mieux les visages visibles dans une image. Pour ce faire, les algorithmes doivent utiliser une définition du visage explicite ou implicite. Celle-ci est le plus souvent construite par apprentissage. Dans le cas explicite, la définition du visage est accessible une fois l'apprentissage terminé.

I.3 Critères utilisés pour mesurer la performance de la localisation de visages

Dans l'analyse et la compréhension d'images par machines, il est usuel d'utiliser les critères comme le taux des détections d_p (détection rate), le taux des mauvaises détections négatives n_f (false négative rate) et le taux des mauvaises détections positives p_f (false positive rate) pour mesurer les performances d'un algorithme. La localisation de visages, étant un sous domaine de l'analyse et la compréhension d'images, utilise aussi ces critères. La suite donne une définition brève de ces critères dans le cas de la localisation de visages [1] :

- **Détection positive :** On appelle « une détection positive » une fenêtre dans l'image qui, selon le détecteur, contient un objet caractéristique.
- **Détection négative :** Réciproquement, on appelle « une détection négative » une fenêtre dans l'image qui, selon le détecteur, ne contient pas d'objet caractéristique.
- **Le taux des (bonnes) détections (positives) :** Le taux des détections d_p est le pourcentage des objets caractéristiques pour lesquels on a une détection positive dans une série d'images.
- **Le taux des mauvaises détections négatives** : Le taux des mauvaises détections négatives n_f est le pourcentage des objets caractéristiques pour lesquels on a une détection négative, dans une série d'images.
- **Le taux des bonnes détections négatives :** Le taux des bonnes détections négatives n_d est le pourcentage des régions qui ne contiennent pas d'objet caractéristique pour lesquelles on a une détection négative dans une série d'images.
- **Le taux des mauvaises détections positives :** Le taux des mauvaises détections positives p_f est le pourcentage des régions qui ne contiennent pas d'objet caractéristique pour lesquelles on a une détection positive dans une série d'images.

L'objectif de la localisation de visages est de maximiser le taux des bonnes détections positives d_p et de minimiser le taux des mauvaises détections positives p_f. Dans les deux cas de figures précédents, on voit bien qu'il est difficile de satisfaire aux deux contraintes en même temps, car souvent les distributions des deux classes sont superposées partiellement. Il

s'agit de trouver un compromis entre le taux des bonnes détections positives p_d et le taux des mauvaises détections positives p_f.

Une courbe souvent utilisée et illustrant très bien la difficulté de satisfaire les deux contraintes mentionnées précédemment s'appelle ROC (Receiver Operating Characteristics). Cette courbe décrit le taux des mauvaises détections négatives n_f en fonction du taux des mauvaises détections positives p_f lorsqu'on fait varier le seuil prédéfini.

I.4 Domaines d'application

La détection de visages est le premier maillon de toute chaîne de traitement de visages. En effet, la majorité des techniques de traitement de visages nécessitent une image normalisée et bien cadrée du visage pour fonctionner. C'est le rôle de la détection de fournir cette image. D'où la détection de visage est abordée dans plusieurs applications telles que :

- En reconnaissance de visages, une base contenant des images de plusieurs personnes est utilisée. Chaque personne est représentée par plusieurs images. Quand l'algorithme reçoit une image d'un visage inconnu, il doit décider s'il s'agit d'une des personnes connues ou non. Cependant, la reconnaissance exige que l'image inconnue donnée en entrée soit sous la même forme que les visages présents dans la base d'images. Cela exige un cadrage contrôlé du visage, qui est assuré par la détection du visage, puis éventuellement d'autres points de repère tels que les yeux, afin de centrer le visage.

- En reconnaissance d'expressions du visage, cette application est cruciale dans l'interaction homme machine. En effet, les mimiques faciales forment une grande part de la communication humaine. Une machine telle qu'un robot par exemple, qui doit communiquer avec des êtres humains, devra être capable de reconnaître les expressions de ses interlocuteurs pour comprendre pleinement leur message. Mais avant de pouvoir reconnaître une expression, il faut être capable de détecter le visage de l'interlocuteur.

- Enfin, la détection de visage peut tout simplement être utilisée pour suivre le déplacement d'une personne et ainsi donner une base solide aux algorithmes de suivi.

La première étape avant de commencer un suivi est d'avoir les coordonnées initiales de l'objet. C'est donc grâce à ce point de départ que les positions futures pourront être estimées. D'autres détections permettront de rendre le suivi plus robuste. Là encore, la détection est non seulement la première étape, mais aussi une étape cruciale au bon fonctionnement du système.

On en déduit que la détection de visages est pour tout traitement du visage, ce que sont les fondations pour une maison. De sa robustesse vont dépendre les performances de tous les autres éléments de la chaîne. Dans le cas idéal, la détection doit être rapide et économique. De plus, elle doit capturer tous les visages présents dans une image et ne doit pas confondre une région de l'arrière-plan avec un visage. C'est là que réside la difficulté.

I.5 Système de détection de visage

Nous vivons actuellement dans l'ère de la technologie et nous essayons d'attribuer les facultés et les capacités humaines aux machines. L'avance scientifique est telle que l'intelligence artificielle est utilisée pour gérer d'une manière optimale des systèmes et équipements complexes afin de les aider à prendre des décisions appropriées. Pour parvenir à un tel résultat on doit passer par la détection des objets et des visages humains dans leur milieu naturel et réel. La raison pour laquelle le processus de détection de visage doit être automatisé sur des machines. Tout processus automatique de détection de visages doit prendre en compte plusieurs facteurs qui contribuent à la complexité de sa tâche, car le visage est une entité dynamique qui change constamment sous l'influence de plusieurs facteurs. Il est donc difficile de présenter un système de détection de visage général et précis avec des détails significatifs. Cependant, la plupart des systèmes comprennent trois étapes (Fig. I.1 [2]). La première étape transforme l'image en entrée pour extraire ou souligner des informations pertinentes afin d'améliorer la précision de la détection. Cette étape est appelée traitement d'image.

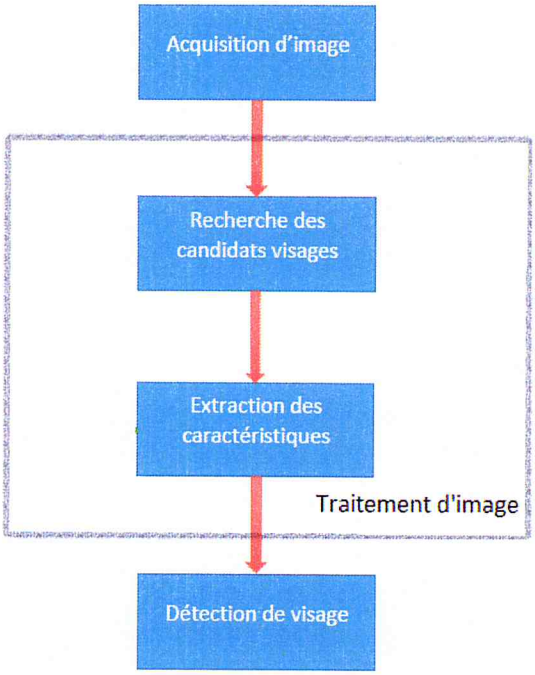


Fig. I.1 Schéma bloc d'un système de détection de visages.

La deuxième étape sélectionne les parties appropriées de l'image transformée pour évaluer davantage. Cette étape est appelée recherche d'image. La troisième et dernière étape évalue les résultats des deux premières étapes et le classifie comme cible ou arrière-plan, visage ou non-visage, et est appelé classification de modèle.

I.6 Problématique

La détection automatique de visage par ordinateur fait face à beaucoup de difficultés. Malgré plusieurs solutions proposées, une solution complète et efficace pour ces problèmes est encore recherchée. Pour gagner en fiabilité, de tels systèmes doivent pouvoir s'affranchir les problèmes suivants qui sont associés à la détection de visage :

1. **La complexité de l'image :** La détection peut être sur des images très complexes avec plusieurs personnes dans la même image, des visages cachés ou à moitié cachés par des objets avec éventuellement des arrière-plans complexes ce qui augmente la difficulté de la détection (Fig. I.2(a)) [2].

2. **Influence des changements d'éclairage :** L'intensité et la direction d'éclairage lors de la prise de vue influent énormément sur l'apparence du visage dans l'image. En effet, dans la plupart des applications courantes, des changements dans les conditions d'éclairage sont inévitables, notamment lorsque les vues sont collectées à des heures différentes, en intérieur ou en extérieur. Etant donnée la forme spécifique d'un visage humain, ces variations d'éclairage peuvent y faire apparaître des ombres accentuant ou masquant certaines caractéristiques faciales (Fig. I.2(b)) [2].

3. **Influence des variations de la pose :** Les changements d'orientation et les changements de l'angle d'inclination du visage engendrent de nombreuses modifications d'apparence dans les images. En effet, les rotations en profondeur engendrent l'occultation de certaines parties du visage comme pour les vues de trois-quarts. D'autre part, elles amènent des différences de profondeur qui sont projetées sur le plan 2D de l'image, provoquant des déformations qui font varier la forme globale du visage. Ces déformations qui correspondent à l'étirement de certaines parties du visage et la compression d'autres régions font varier aussi les distances entre les caractéristiques faciales (Fig. I.2(c)) [2].

4. **Présence ou absence des composants structuraux :** Les caractéristiques faciales telles que la barbe, la moustache, et des lunettes peuvent ou ne peuvent pas être présentes et il y a beaucoup de variabilités parmi ces composants comprenant la forme, la couleur, et la taille. De plus, si celles-ci apparaissent, elles peuvent cacher autres caractéristiques faciales de base ((Fig. I.3(d) [2].

5. **Influence des occultations :** Un visage peut être partiellement masqué par des objets ou par le port d'accessoires tels que des lunettes, un chapeau, une écharpe. Les occultations peuvent être intentionnelles ou non. Dans le contexte de la vidéosurveillance, il peut s'agir d'une volonté délibérée d'empêcher la reconnaissance. Il est clair que la reconnaissance sera d'autant plus difficile que peu d'éléments discriminants seront simultanément visibles (Fig. I.2(e)) [2].

6. **Influence des expressions faciales :** Les visages sont des éléments non rigides. Les expressions faciales véhiculant des émotions, combinées avec les déformations induites par la parole, peuvent produire des changements d'apparence importants, et le nombre de configurations possibles devient trop important pour que celles-ci soient décrites in extenso de façon réaliste (Fig. I.2(f)) [2].

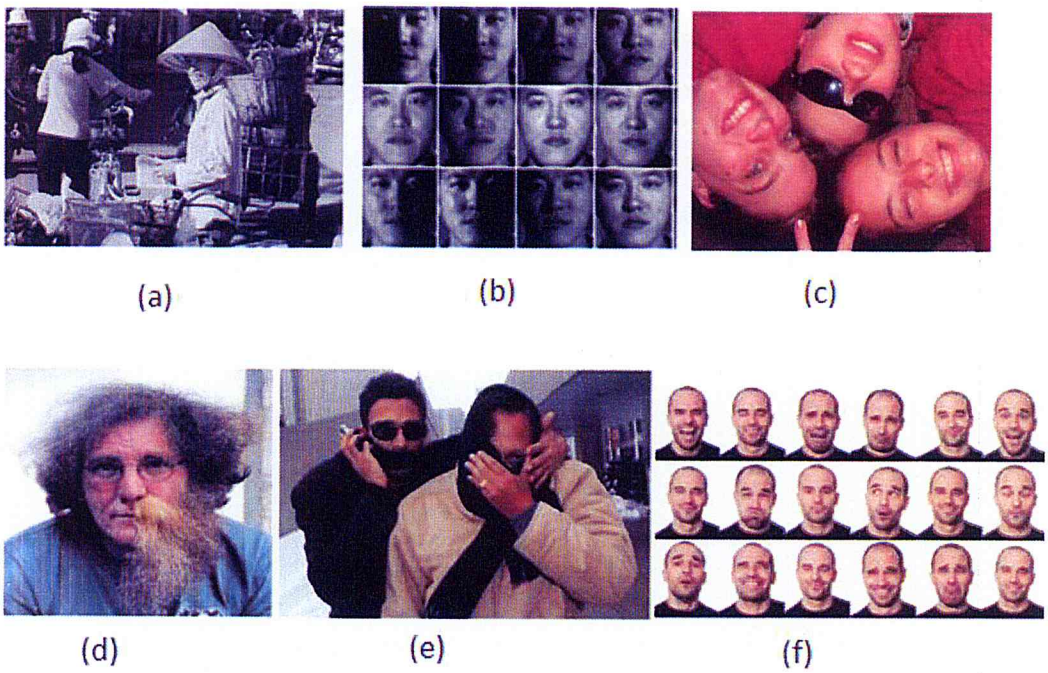


Fig. I.2 Certaines difficultés de détection.

Un bon système de détection de visage doit fournir des solutions fiables pour surmonter les problèmes exposés ci-dessus. Plusieurs méthodes de détection de visage ont été proposées dans la littérature [3]. Dans la partie suivante, nous allons passer en en revue quelques techniques de détection de visage parues dans la littérature.

I.7 Méthodes de détections réparties en catégories

Les premiers efforts de détection de visage remontent au début des années 1970, où les techniques heuristiques et anthropométriques ont été utilisées [3]. Ces techniques sont en grande partie rigides en raison de diverses hypothèses, comme : fond uni, visage frontal, un exemple type la photo du passeport. Pour ces systèmes, toute modification des conditions de

l'image signifie un réglage fin sinon il faudrait reconcevoir tout le système. En raison de ces problèmes, la recherche a stagné jusqu'en 1990 [4]. Lorsque la reconnaissance de visage et les systèmes de codage vidéo ont commencé à devenir une réalité, les chercheurs ont présentés des mécanismes de segmentation robustes, notamment ceux qui utilisent le mouvement, la couleur, et les informations généralisées. L'utilisation des statistiques et des réseaux de neurones a également permis de détecter le visage dans des scènes encombrées et à différentes distances de l'appareil photo. Avant 2000 des centaines de méthodes de détection de visage ont fait leurs apparitions, et ont été bien étudiés par Yang et al. [5] et Hjelmas al. [6].

I.7.1 Méthodes basées sur la connaissance (Knowledge-based methods)

Ces méthodes se basent sur la connaissance des différents éléments qui constituent un visage ainsi que sur des relations qui existent entre eux. Chiang et al. [7], ont employé une méthode hiérarchique basée sur la moyenne et un sous échantillonnage pour détecter les positions relatives de différents éléments clés du visage tels que la bouche, le nez et les yeux. Ces caractéristiques sont ensuite utilisées pour la classification « visage » ou « non-visage ».

Le principe de la méthode comprend trois étapes.

1ère étape : Tous les candidats possibles de visage sont trouvés en balayant une fenêtre sur l'image d'entrée et en appliquant un ensemble de règles à chaque région de l'image. Les règles à ce niveau sont des descriptions générales du visage.

Les règles codées qui sont utilisées pour localiser des candidats de visage dans la plus basse résolution sont :

- **1er règle** : la partie centrale du visage (les parties foncées dans Fig. I.3 [5]) a neuf cellules avec une intensité fondamentalement uniforme.
- **2ième règle** : la pièce ronde supérieure d'un visage (les parties gris claires dans Fig. I.4 [5]) a une intensité uniforme fondamentalement.
- **3ième règle** : est-ce que la différence entre les valeurs grises moyennes de la partie centrale et la partie ronde supérieure est significative ?

L'image au niveau de la plus basse résolution est examinée pour détecter des candidats de visages et ceux-ci doivent être encore examinés à des résolutions plus fines.

2ème étape : L'égalisation locale d'histogramme est appliquée sur les candidats obtenus du visage suivi par la détection de contours.

3ème étape : Les régions des candidats sont alors examinées avec un autre ensemble de règles qui répondent aux organes faciaux tels que les yeux et la bouche. Les règles à des niveaux plus bas se fondent sur les détails des composants faciaux.

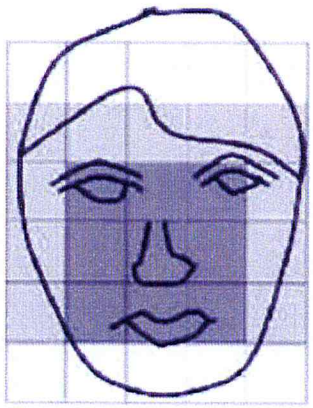


Fig. I.3 Le modèle type exploité par la méthode de *Yang et al.* [5].

Le problème dans ce type de méthode est qu'il est difficile de bien définir de manière unique un visage. En effet, certains visages seront ratés avec un taux de faux positive croissant.

I.7.2 Approches des entités invariantes (Feature invariant approaches)

Ces algorithmes visent à trouver les caractéristiques structurelles invariables qui existent même lorsque la pose, l'angle de vue, ou la condition d'éclairage changent, et les utiliser pour localiser les postions des visages. Les caractéristiques souvent utilisées sont : la forme, la texture, la couleur de peau, le contour…

Il existe principalement trois familles d'approches basées sur les caractéristiques invariables. Les premières utilisent la propriété de la peau humaine pour capter des régions contenant des visages. Les méthodes de la deuxième famille visent à détecter les caractéristiques de visage; elles consistent à localiser les cinq caractéristiques (deux yeux, deux narines, et la jonction

nez/lèvre) pour décrire un visage typique. La troisième comprend les méthodes hybrides qui combinent une multitude de caractéristiques de natures différentes.

I.7.3 Modèle de correspondance (Template matching methods)

L'idée principale de ces approches est de créer des modèles standards capables de décrire un visage ou une partie du visage. Puis, effectuer la corrélation entre l'image d'entrée et les modèles calculés pour repérer des visages dans l'image. Dans ce cas, des modèles représentatifs de visage sont construits au préalable. Le problème de la détection de visage est ramené à un problème de mise en correspondance des formes: on vérifie à chaque endroit de l'image si une fenêtre est un candidat de visage en comparant la différence entre celle-ci et les modèles de visage. Malgré le fait que les visages ont la même structure, ils peuvent être de différentes tailles et positions, ...etc. Donc, la construction des modèles de visage est très importante. D'autre part, les conditions d'illumination affectent la distribution de couleur de l'image et donc elles introduisent un bruit au contour. Ceux-ci constituent les défis du problème.

En général, la mise en correspondance est basée sur des fonctions de corrélations croisées de dimensions deux entre une fenêtre de l'image et le modèle [8]. Des modèles de visage normalisés sont soit prédéfinis manuellement par un spécialiste, soit paramétrés automatiquement par des fonctions.

I.7.4 Méthodes fondées sur l'apparence (Appearance-based methods)

Ces méthodes utilisent le même principe que présenté au point précédent mais se basent sur des modèles appris à partir d'un ensemble d'essai. Ces méthodes présentent l'avantage de s'exécuter très rapidement mais demandent un long temps d'entraînement.

Le principe de ces méthodes est de considérer le problème de la détection de visage comme un problème de classification : Ici, il s'agit de classer un modèle capturé dans l'une des deux classes : classe de visages et classe de non-visages. Les techniques utilisent l'analyse statistique et l'apprentissage automatique pour construire des machines capables de séparer les visages des non-visages. Les réseaux de neurones, les machines à vecteurs de support (SVM)

[6], les classifier bayésiens, les modèles de Markov cachés (HMM) **[2]** sont parmi les techniques d'apprentissage automatique les plus souvent utilisées.

Bien que certaines méthodes récentes, basées sur des caractéristiques invariantes aient amélioré la capacité face à l'incertitude, la plupart des méthodes sont encore limitées pour détecter les visages frontaux. Il y a toujours un besoin de techniques qui peuvent s'exécuter dans les scénarios les plus hostiles tels que la détection de multiples visages devant un fond complexe. Les méthodes appartenant à cette catégorie ont montré de bons résultats par rapport aux trois autres types de méthodes [10]. On peut citer parmi celles-ci, la méthode basée sur les réseaux de neurones de Rowley et al. [11], la méthode de Schneiderman et Kanade [12] basée sur un classifieur de Bayes naïf ainsi que le fameux algorithme de Viola et Jones [13] fonctionnant en temps réel, et ce dernier sera détaillé ci-dessous. Le Tableau I.1 regroupe les quatre catégories de méthodes citées auparavant.

Tableau I.1 Avantages et inconvénients des méthodes de détection de visage dans une seule image [9].

catégories	Avantages	Inconvénients
Méthodes basées sur la connaissance (Knowledge-based methods)	-Rapidité	- difficile de bien définir de manière unique un visage. certains visages sont ratés avec un taux de faux positive élevé.
Approches invariantes des entités (Feature invariant approaches) • caractéristiques du visage • texture • couleur de peau • plusieurs fonctionnalités	-Rapidité - Détection de la peau efficace	- Détection des yeux peu robuste - Conflits avec l'arrière-plan
Modèle de correspondance (Template matchingmethods) • modèles de visage prédéfinis • modèle déformable	- la conception simple - Mesure de similarités	Recherche multi-échelle - Filtrage des multiples détections - Faible précision - Modèle représentatif
Modèle de correspondance (Template matchingmethods) • Eigenface • distibition basée • réseau neuronal • support vecteur machine SVM • classifieur bayesien naïf • markov caché mpdel HMM	- Gabarit moins crucial - Moins sensible à l'éclairage ; - Mesure de similarité -simple à mettre en œuvre (SVM) ; - Apprentissage automatique ; -Capacité de généralisation (réseau neuronal)	faible performance -gourmande en temps et en précision (SVM). -Faible précision - Recherche multi-échelle (réseau neuronal)

I.8 Algorithme de Viola et Jones

Une avancée majeure dans le domaine a été réalisée par les chercheurs Paul Viola et Michael Jones en 2001 [17].

Ces derniers ont proposé une méthode basée sur l'apparence (Appearance-basedmethods).

La méthode de Viola et Jones est une méthode de détection d'objet dans une image numérique, elle fait partie des toutes premières méthodes capables de détecter efficacement et en temps réel des objets dans une image. Inventée à l'origine pour détecter des visages, elle peut également être utilisée pour détecter d'autres types d'objets comme des voitures ou des avions. La méthode de Viola et Jones est l'une des méthodes les plus connues et les plus utilisées, en particulier pour la détection de visages et la détection de personnes.

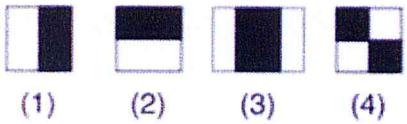
En tant que procédé d'apprentissage supervisé, la méthode de Viola et Jones nécessite de quelques centaines à plusieurs milliers d'exemples de l'objet que l'on souhaite détecter, pour entraîner un classifieur. Une fois son apprentissage réalisé, ce classifieur est utilisé pour détecter la présence éventuelle de l'objet dans une image en parcourant celle-ci de manière exhaustive, à toutes les positions et dans toutes les tailles possibles.

Principe :

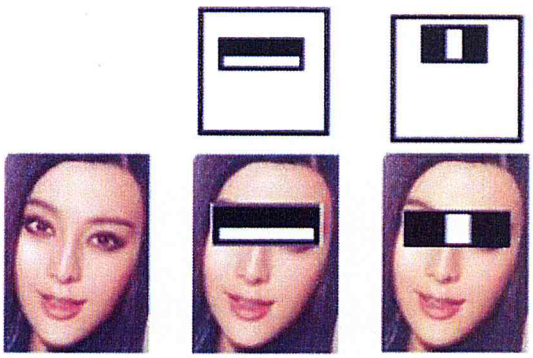
La méthode de Viola et Jones consiste à balayer une image à l'aide d'une fenêtre de détection de taille initiale 24 pixels par 24 pixels (dans l'algorithme original) et de déterminer si un visage y est présent. Lorsque l'image a été parcourue entièrement, la taille de la fenêtre est augmentée et le balayage recommence, jusqu'à ce que la fenêtre fasse la taille de l'image. L'augmentation de la taille de la fenêtre se fait par un facteur multiplicatif de 1.25. Le balayage, quant à lui, consiste simplement à décaler la fenêtre d'un pixel. Ce décalage peut être changé afin d'accélérer le processus, mais un décalage d'un pixel assure une précision maximale.

Cette méthode est une approche basée sur l'apparence, qui consiste à parcourir l'ensemble de l'image en calculant un certain nombre de caractéristiques dans des zones rectangulaires qui se chevauchent. Elle a la particularité d'utiliser des caractéristiques très simples mais très nombreuses. Les caractéristiques utilisées par Viola et Jones sont les caractéristiques pseudo-Haar aussi connues sous le nom de Haar-Like (Fig. I.4) [2]. Elles sont calculées par la

différence entre la somme de tous les pixels dans le rectangle blanc et la somme de tous les pixels dans le rectangle noir.



(a) caractéristiques pseudo-Haar



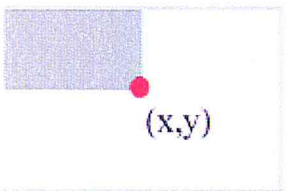
(b) Détection de visage: application des caractéristiques pseudo-Haar.

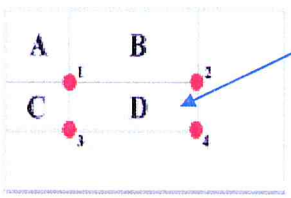
Fig. I.4 Les 4 types de rectangles utilisés pour l'extraction des caractéristiques du visage et leur application.

Il est possible de calculer très rapidement les caractéristiques de Haar à l'aide des images intégrales. Une image intégrale est construite à partir de l'image d'origine, et de même taille qu'elle. Elle contient en chacun de ses points la somme des pixels situés au-dessus et à gauche du pixel courant, l'image intégrale I est définie à partir de l'image d'origine *f(x,y)* par :

$$I(x,y) = \Sigma\Sigma f(m,n) \qquad (I.1)$$

Une fois le calcul de l'image intégrale effectué (**Fig. I.5**), la somme des pixels dans n'importe quel rectangle situé dans l'image peut se calculer en seulement 3 opérations et 4 accès à l'image intégrale. Une caractéristique de Haar à deux rectangles peut alors être déterminée en seulement 6 accès (2 points sont partagés) à l'image, et une caractéristique à 3 rectangles en seulement 8 accès.





La somme des pixels est gale à (1+4-(2+3).

Fig. I.5 Exemple d'image intégrale.

La méthode de Viola et Jones est basée sur une approche par recherche exhaustive sur l'ensemble de l'image, qui teste la présence de l'objet dans une fenêtre à toutes les positions et à plusieurs échelles. Cette approche est cependant extrêmement coûteuse en calcul. L'une des idées-clés de la méthode pour réduire ce coût réside dans l'organisation de l'algorithme de détection en une cascade de classifieurs. Appliqués séquentiellement, ces classifieurs prennent une décision d'acceptation, la fenêtre contient l'objet et l'exemple est alors passé au classifieur suivant, ou de rejet, la fenêtre ne contient pas l'objet et dans ce cas l'exemple est définitivement écarté. L'idée est que l'immense majorité des fenêtres testées étant négatives, il est avantageux de pouvoir les rejeter avec le moins possible de calculs. Ici, les classifieurs les plus simples, donc les plus rapides, sont situés au début de la cascade, et rejettent très rapidement la grande majorité des exemples négatifs.

Une des limitations de la méthode de Viola et Jones est son manque de robustesse à la rotation, et sa difficulté à apprendre plusieurs vues d'un même objet. En particulier, il est difficile d'obtenir un classifieur capable de détecter à la fois des visages de face et de profil. Viola et Jones ont proposé une amélioration qui permet de corriger ce défaut [17], qui consiste à apprendre une cascade dédiée à chaque orientation ou vue, et à utiliser lors de la détection un arbre de décision pour sélectionner la bonne cascade à appliquer.

Plusieurs autres améliorations ont été proposées par la suite pour apporter une solution à ce problème [18, 19].

I.9 Problématique méthodes de détection de visage classique

Les méthodes classiques ne sont pas robustes car elles sont imprécises dans certains cas particuliers. La détection de visage est un sujet difficile, notamment dû à la grande variabilité d'apparence des visages dans des conditions non contraintes:

- Variabilité intrinsèque des visages humains (couleur, taille, forme) ;
- Présence ou absence de caractéristiques particulières (cheveux, moustache, barbe, lunettes...) ;
- Expressions faciales modifiant la géométrie du visage ;
- Occultation par d'autres objets ou d'autres visages ;
- Orientation et pose (de face, de profil) ;

- Conditions d'illumination et qualité de l'image ;
- Présence de photos de visages qui seront prises comme des visages de personnes en vrais.

Pour éviter tous les problèmes cités au-dessus, il a été démontré que d'excellents résultats de détection de visage peuvent être obtenus grâce à la localisation de points de repère du visage appelés aussi les points de repères faciaux ou « landmarks ». Une fois que le visage est localisé, il est nécessaire de trouver des points caractéristiques du visage. Ces points vont être les coins des yeux, le nez, la bouche, les sourcils, …. Ces caractéristiques sont régulièrement appelées facial landmarks, que l'on peut traduire par repères faciaux.

Un bon système de reconnaissance faciale repose en grande partie sur une bonne localisation de points de repère du visage. Cette localisation doit être invariante aux poses et aux conditions d'éclairage. Elle ne doit donc de préférence pas utiliser l'information texturale.

I. 10 Conclusion

Nous avons consacré la première partie de ce chapitre à la présentation des notions de base de la détection de visage et ses applications. Dans la seconde partie, nous avons passé en revue quelques méthodes classiques développées dans la littérature particulièrement la méthode de Viola et Jones qui est très connue et largement utilisée dans les applications de traitement d'image et vidéo. Dans le chapitre II, nous aborderons particulièrement des méthodes de détection de visage basées sur l'alignement des points de repère du visage.

Chapitre II
Alignement du visage

II.1 Introduction

L'alignement de points caractéristiques du visage est un domaine de recherche actif, à la frontière entre l'apprentissage statistique et la vision par ordinateur. Il vise à localiser précisément un ensemble de marqueurs (coins de bouche, des yeux, bout du nez). En effet, depuis une quinzaine d'années, les recherches sur l'analyse automatique des visages et sur l'alignement de points caractéristiques ont connu des avancées formidables et les technologies actuelles sont suffisamment matures pour être exploitées dans des applications et des services commerciaux tels que la biométrie, l'interaction homme/robot, l'analyse marketing et comportementale.

Dans ce chapitre, nous aborderons l'intérêt apporté par les méthodes d'alignement des traits du visage et nous décrirons brièvement quelques techniques élaborées.

II.2 Définition

L'alignement consiste à trouver pour chaque forme une translation, une mise à l'échelle et un angle de rotation qui minimisent la différence avec les autres (somme des distances point à point). Pour le visage, il s'agit d'extraire les trais permanents du visage à savoir : les yeux, les sourcils les lèvres, le nez et la bouche. Puis pour chacun des traits considérés, un modèle paramétrique spécifique capable de rendre compte de toutes les déformations possibles est défini. Lors de la phase d'initialisation, des points caractéristiques du visage sont extraits (coins des yeux et de la bouche par exemple) et servent de points d'ancrage initiaux pour chacun des modèles. Dans la phase d'évolution, chaque modèle est déformé afin de coïncider au mieux avec les contours des traits présents sur le visage analysé. Cette déformation se fait par maximisation d'un flux de gradient (de luminance et/ou de chrominance) le long des contours définis par chaque courbe du modèle [20].

La définition de modèles permet d'introduire naturellement une contrainte de régularisation sur les contours recherchés. Néanmoins, les modèles choisis restent suffisamment flexibles pour permettre une extraction réaliste des contours des yeux, des sourcils et de la bouche. L'extraction précise des contours des principaux traits du visage constitue la première étape d'un ensemble d'applications multimédia.

II.3 Les défis de la localisation historique

Il existe quatre principaux défis dans la localisation des traits (points de repère) faciaux. Ce sont les suivants [21] :

- **Variabilité:** Les apparences des repères diffèrent en raison de plusieurs facteurs tels que l'occlusion partielle, la pose, l'illumination et la résolution, également en raison de facteurs tels que la variabilité du visage entre les individus. Les points de repère faciaux peuvent parfois être seulement partiellement observés en raison des mouvements de la main ou de l'auto-occlusion en raison de rotations de tête étendues ou d'occlusions de cheveux.

  Les détections de repères faciaux sont également difficiles à cause des artefacts d'éclairage et des expressions faciales. Un algorithme de localisation de repères faciaux qui fournit les points cibles à un moment de manière efficace et qui fonctionne bien sur toutes les variations des visages n'a pas encore été réalisable.

- **Précision et nombre de points de repère:** En fonction de l'application prévue, le nombre de points de repère et leur précision varient. Par exemple, dans les tâches de reconnaissance faciale ou de détection de visage, Les cinq repères primaires comme le coin de deux bouches, les yeux et le nez peuvent être adéquats. Les coordonnées des repères localisés donnent lieu à un certain nombre de propriétés géométriques telles que l'angle et la distance entre les composants faciaux.

- **Absence de jeu de données globalement accepté et sans erreurs:** la plupart des jeux de données fournissent des annotations avec des annotations différentes et la précision de leur point de référence est discutable. La précision de l'algorithme de localisation des repères dépend en grande partie de l'ensemble de données utilisé pour la formation. Chaque algorithme utilise des ensembles de données différents

pour former et évaluer les performances, de sorte qu'il est difficile de comparer les algorithmes.

- **Conditions d'acquisition:** Les conditions d'acquisition, telles que la résolution, l'encombrement de fond, l'éclairage, peuvent affecter la performance de localisation du point de repère. Les localisateurs de repère formés dans une base de données ont généralement des performances inférieures lorsqu'ils sont testés sur une autre base de données.

II.4 Techniques de localisation de repères faciaux

La détection des éléments faciaux correspond généralement à la localisation des points caractéristiques, comme les yeux, le nez et la bouche, on peut s'intéresser également à la détection des coins des yeux, des sourcils, de l'iris et des points de contour du menton.
Les approches proposées dans la littérature sont très nombreuses et dépendent fortement du contexte applicatif.
La plupart des approches opèrent dans le rectangle encadrant le visage qu'on appelle aussi boîte englobante, une fois celui-ci détecté. L'objectif est alors d'aligner les visages pour détecter les points de repères. Dans le cas général, les algorithmes doivent être robustes aux défis de la localisation et aux erreurs de centrage de la boîte englobante.
Les approches de détection d'éléments faciaux peuvent être schématiquement divisées en deux catégories : les approches basées sur la recherche contrainte de caractéristiques et les approches basées sur des modèles déformables.

- Les méthodes de la première catégorie cherchent à détecter certains éléments faciaux et cela indépendamment les uns des autres.
- La seconde catégorie regroupe, quant à elle, les techniques d'alignement de visages se basant sur une modélisation globale du visage.

Les méthodes de la première catégorie appliquent des traitements locaux par filtrage ou corrélation, pour localiser des éléments faciaux candidats parmi lesquels est sélectionnée la meilleure combinaison par rapport à un modèle géométrique.
Les approches de la seconde catégorie tentent de mettre en correspondance itérativement une grille, un graphe ou un modèle déformable sur le visage.
La position des nœuds de la grille ou des points de support du modèle déformable correspondent aux éléments faciaux lorsque l'algorithme a convergé.

II.4.1 Approches basées sur la recherche contrainte de caractéristiques

Les premières méthodes ayant été proposées reposent sur une analyse bas-niveau de la couleur ou de l'intensité de l'image du visage. Elles sont peu robustes aux variations d'éclairage, aux variations d'expressions faciales et à la présence d'occultations partielles.

Parmi ces méthodes, de très nombreuses appliquent des techniques de segmentation reposant sur le filtrage de la teinte particulière des zones du visage comme les yeux [24, 25, 26, 27, 28], les narines [29] et les lèvres [30, 31, 32, 28, 33]. Certaines approches accumulent les valeurs d'intensité de chaque pixel le long des lignes et des colonnes de l'image afin d'obtenir des courbes d'accumulation horizontales et verticales, dont l'analyse des maxima locaux permet de repérer les lèvres, les yeux et le nez [34, 35, 36].

D'autres reposent sur la recherche d'éléments géométriques à partir des contours, notamment en utilisant la transformée de Hough (des ellipses pour rechercher les yeux [33], des hyperboles pour les lèvres [37] ou des cercles pour les iris des yeux [38, 39]) ou encore les contours actifs, des formes géométriques déformables dynamiquement vers les formes recherchées, afin d'obtenir par exemple les contours des lèvres [40, 41

D'autres méthodes ne reposent pas sur des traitements purement bas-niveau, mais proposent d'utiliser des détecteurs spécialisés pour chaque élément à rechercher, mettant en œuvre des classifieurs qui prennent en compte les zones d'images autour de chaque élément, à la manière des détecteurs de visages basés image. Les éléments faciaux sont alors recherchés indépendamment et les positions candidates sont ensuite filtrées grâce à des modèles géométriques intégrant les contraintes morphologiques du visage. *Reinders et al.* [43] ont présenté une approche utilisant un MLP (chapitre III) entraîné sur les orientations du gradient afin de détecter approximativement les positons des yeux et ensuite quatre MLPs pour détecter les micro-caractéristiques. *Leung et al.* [44] ont appliqué un ensemble de filtres multi-orientations et multi-échelles basés sur des dérivées Gaussiennes pour localiser quatre points caractéristiques du visage. Ils apprennent les configurations correctes des éléments faciaux à partir de la distribution Gaussienne des distances mutuelles. *Yow et Cipolla* [45] ont utilisé aussi des filtres approximant les dérivées de Gaussiennes d'ordre deux pour localiser les éléments faciaux. Les contours autour de ces points d'intérêt sont organisés en paires, la méthode de regroupement et de fusion en visage candidat étant basée sur le réseau de croyance (belief networks). *Moghaddam et al.* [46] ont présenté une méthode qui modélise les éléments faciaux en appliquant un ACP (Eigenfeatures). Durant la recherche, chaque zone d'image, extraite dans une fenêtre de dimensions adaptées à chaque élément facial, est

projetée dans l'espace propre (EigenfeatureSpace) de chaque élément facial puis reconstruite. L'erreur résiduelle entre la zone d'image et sa reconstruction permet de construire une distance (Distance From Feature Space, DFFS) dont le minimum permet de repérer l'élément facial correspondant. La distribution Gaussienne des éléments détectés permet de valider les configurations correctes à l'aide d'un modèle probabiliste. *Feris et al.* [47] quant à eux ont proposé une approche hiérarchique à deux niveaux utilisant des réseaux d'ondelettes de Gabor (Gabor Wavelet Networks, GWN). Le GWN de premier niveau est entraîné pour localiser le visage et les positions approximatives des éléments faciaux (yeux, coins de la bouche et narines). Le GWN de second niveau affine alors leur localisation.

Parmi les méthodes basées sur la recherche contrainte de caractéristiques, nous détaillons les premiers modèles proposés pour les yeux, les sourcils et les lèvres.

II.4.1.1 Modèle de bouche :

Plusieurs modèles paramétriques ont déjà été proposés pour modéliser le contour des lèvres.

Tian et al. [22] Utilise un modèle constitué de deux paraboles (Fig. II.1(a)). [22]

C'est simple à calculer mais la précision obtenue est très limitée.

D'autres auteurs ont proposé de modéliser le contour supérieur des lèvres à l'aide de deux paraboles au lieu d'une (Fig. II.1(b)). [22]

Ou encore d'utiliser des quartiques (Fig. II.1(c)). [22]

Un gain en précision a été obtenu, néanmoins tous ces modèles sont encore limités par leur trop grande rigidité, en particulier dans le cas d'une bouche non symétrique.

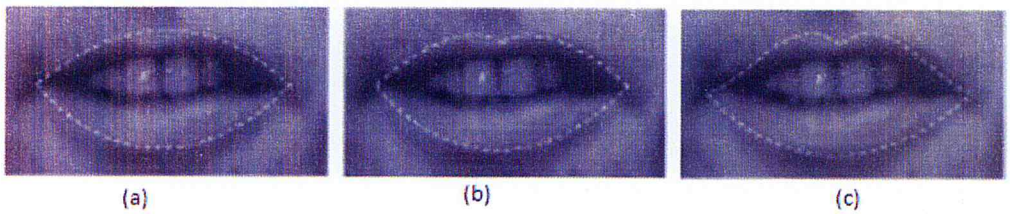


Fig.II.1 (a) modèle à 2 paraboles, (b) modèle à 3paraboles, (c) modèle à base de quartiques.

Le choix d'un modèle adapté pour modéliser les lèvres est très délicat car la forme des lèvres est très variable. L'utilisation d'un modèle a priori lors de la phase de segmentation permet une régularisation des contours recherchés. Mais si le modèle choisi n'est pas bien adapté, le

résultat de la segmentation ne sera pas de bonne qualité. Le modèle proposé par Tian et al. [22] est composé de 5 courbes indépendantes, chacune d'entre elles décrit une partie du contour labial (Fig. II.2. Entre Q_2 et Q_4, l'arc de Cupidon est décrit par une ligne brisée tandis que les autres portions du contour sont décrites par des courbes polynomiales cubiques. A chaque cubique la dérivée au point Q_2, Q_4 ou Q_6 nulle est imposée.

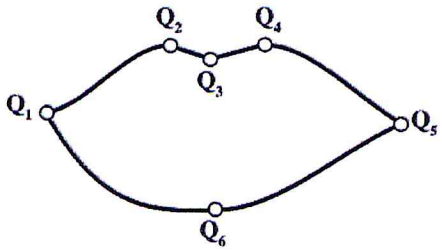


Fig.II.2 Modèle proposée par Tian et al. [22] Pour la bouche.

II.4.1.2 Modèle pour les yeux et les sourcils

Le modèle le plus courant proposé jusqu'ici pour les yeux est un modèle à base de paraboles pour la modélisation de la frontière entre les yeux et les paupières et un modèle circulaire pour la modélisation de l'iris [20] [23]. L'étude plus précise des contours des yeux sur la base d'images ORL [21] a montré que le contour associé à la paupière supérieure ne présentait pas nécessairement une symétrie verticale. Sur la Fig. II.3, le cas d'un œil à paupière supérieure non symétrique est illustré. L'œil est mal approché par une parabole (Fig.II.3(a)) mais qui est bien délimité par une courbe de Bézier plus flexible (Fig.II.3(b)). En revanche, le contour de la paupière inférieure présente une telle symétrie d'où le choix d'une parabole. L'idée est de choisir pour chaque contour la courbe la plus adaptée mais aussi la plus simple possible.

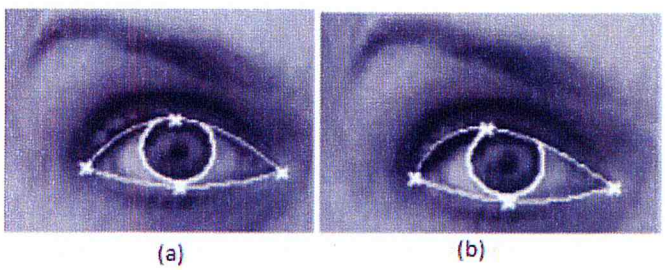


Fig.II.3 Œil à paupière supérieure non symétrique : (a) modèle à deux paraboles, (b) modèle à une courbe de Bézier pour le contour supérieur et une parabole pour le contour inférieur.

En ce qui concerne les sourcils, ceux-ci sont modélisés la plupart du temps par deux lignes

brisées passant par les deux coins et un point milieu ce qui est très rudimentaire [20]. Généralement un modèle paramétrique est utilisé (Fig.II.4) : Son principe est comme suit :

- Pour chaque œil : un cercle pour l'iris (éventuellement incomplet si l'œil est semi-ouvert);
- Pour le contour inférieur, une parabole définie par trois points $\{P_1, P_2, P_4\}$;
- Pour le contour supérieur, une courbe de Bézier à trois points de contrôle $\{P_1, P_2, P_3\}$;
- Pour le cas d'un œil fermé, une droite passant par P_1 et P_2 est employée ;
- Pour les sourcils : une courbe de Bézier à trois points de contrôle $\{P_5, P_6, P_7\}$
- Pour le contour inférieur (on se limite à ce contour).

De manière générale, si on note $A(x_a, y_a)$, $B(x_b, y_b)$ et $C(x_c, y_c)$ les coordonnées de trois points de contrôle, les coordonnées (x,y) de la courbe de Bézier associée sont définies par [20] :

$$x = \frac{(1-t)2x_a + 2t(1-t)(x_c + x_a)}{2tx_b} \quad \text{(II.1)}$$

$$y = \frac{(1-t)2y_a + 2t(1-t)(y_c + y_a)}{2ty_b} \quad \text{(II.2)}$$

où t est un paramètre appartenant à [0 , 1]

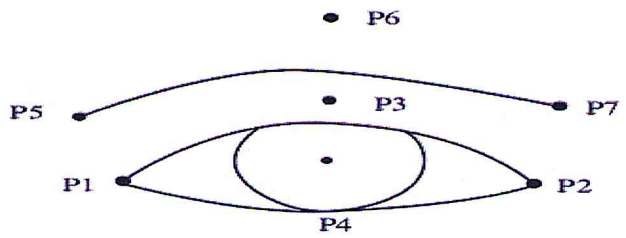


Fig.II.4 Modèles choisis pour l'œil et le sourcil.

Le processus d'alignement de visage identifie la structure géométrique des visages dans les images, et tente d'obtenir un alignement canonique du visage basé sur la base des opérations de translation, changement d'échelle et la rotation. Il existe de nombreuses formes d'alignement de visage, certaines méthodes imposent un modèle (prédéfini) et appliquent une transformation à l'image d'entrée telle que les points de repère sur la face d'entrée correspond à des points de repère sur le modèle. Autres méthodes plus simples (comme celle discutée dans ce chapitre), les méthodes s'appuient uniquement sur les repères de visage d'entrée (en

particulier, les régions de l'œil) pour obtenir un normalisé rotation, translation et échelle de représentation du visage.

Alignement du visage et points de repère du visage

Les étapes avant la localisation des points de repère du visage sont :

- Centrer localiser le visage dans l'image ;
- Faire pivoter le visage de telle façon que les yeux se trouvent sur une ligne horizontale.
- Toute les images dans la base de donnée doivent être de tailles identiques.

II. 4.2 Approches basées sur des modèles déformables

Les approches les plus populaires reposent sur une modélisation statistique des distributions possibles d'un ensemble de points caractéristiques du visage (constituant une forme). *Sozou et al.* [48] ont introduit les modèles à distribution de points (Point Distribution Model, PDM), afin de former le prototype d'une forme moyenne doté de modes de variation combinables, appris à l'aide d'une ACP, sur un ensemble d'apprentissage d'instances de la forme Exploitant le PDM. *Cootes et al.* [49], ont proposé les modèles actifs de forme (Active Shape Model, ASM). L'algorithme ASM cherche à mettre en correspondance la forme sur une image de visage en alternant itérativement entre deux étapes qui consistent à chercher localement autour de chaque point de la forme, la meilleure position (sur le contour le plus proche) et à mettre à jour les paramètres de la forme obtenue. La mise en correspondance des points étant locale et reposant sur le gradient, l'approche ASM est sensible à l'initialisation en position et en échelle du modèle. *Cootes et al.* [50] ont proposé de modéliser conjointement les variations possibles de forme et de texture (intensité des pixels), en introduisant les Modèles d'Apparences Actifs (Active AppearenceModel, AAM). Un PDM est dans un premier temps construit sur la base des formes. Les textures sont alors déformées pour s'aligner sur la forme moyenne, via des opérations de déformation (warping) de triangles issus de la triangulation de Delaunay. Un ACP est alors appliqué sur des vecteurs concaténant les vecteurs de forme et de texture correspondants à chaque exemple, permettant ainsi d'estimer l'apparence moyenne et ses principaux modes de variations. En phase de recherche, à partir d'une position initiale de la forme sur le visage (placée en fonction de la boîte englobante), le vecteur d'apparence moyen est projeté dans l'espace des apparences, donnant un vecteur de paramètres qui est ensuite itérativement modifié pour que l'erreur résiduelle entre l'apparence reconstruite correspondante et l'image originale soit minimisée.

II.6 Conclusion

Cette revue de littérature dans ce chapitre, nous a permis d'expliquer succinctement comment détecter automatiquement les repères faciaux d'un visage. Plusieurs approches ont été introduites, ces approches vont nous servir dans le cadre de ce mémoire à atteindre notre objectif qui est alignement des repères faciaux pour une détection du visage précise et robuste. Les récents progrès dans le domaine des réseaux de neurones artificiels (plus connu actuellement sous le nom d'apprentissage profond) ont permis d'améliorer l'état de l'art dans plusieurs domaines de la vision par ordinateur en offrant une possibilité de s'attaquer à des problèmes qui étaient difficilement traitables par les méthodes de détection de visage automatique conventionnelles. Ainsi, dans le cadre de ce mémoire, nous étudions la question suivante : comment des techniques d'apprentissage profond peuvent telles aider dans la détection précise du visage en prenant en compte toutes les contraintes possibles. Dans le chapitre suivant, nous aborderons le concept des réseaux de neurones convolutionels profonds et les différentes architectures proposées.

Chapitre III
Réseaux de neurones convolutionnels

III.1 Introduction

Dans les chapitres précédents, nous avons majoritairement évoqué les algorithmes classiques de détection de visage et d'alignement. Ce chapitre, a pour objectif de présenter un type particulier de réseau de neurones appelé réseau de neurones convolutif (CNN). Nous rappellerons l'intérêt des réseaux de neurones et nous donnerons quelques définitions et équations utilisées tout au long de ce mémoire.

III.2 Définition des réseaux de neurones

Un réseau neuronal est l'association, en un graphe plus ou moins complexe, d'objets élémentaires, les neurones formels. Les principaux réseaux se distinguent par l'organisation du graphe (en couches, complets…), c'est-à-dire leur architecture, son niveau de complexité (le nombre de neurones, présence ou non de boucles de rétroaction dans le réseau), par le type des neurones (leurs fonctions de transition ou d'activation) et enfin par l'objectif visé : apprentissage supervisé ou non, optimisation, systèmes dynamiques. La Fig.III.1[52] montre un schéma comportant les organes principaux d'un neurone formel

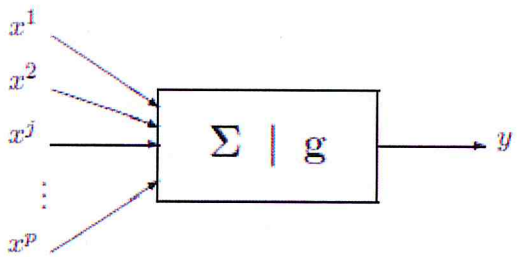


Fig.III.1 Représentation d'un neurone formel.

III.2 Réseaux de neurones

Nous présenterons dans un premier temps l'entité de base d'un réseau de neurones. Ensuite, nous présenterons le perceptron multicouche

III.2.1 Le Neurone formel

De façon très réductrice, un neurone biologique est une cellule qui se caractérise par :

- des synapses,
- les points de connexion avec les autres neurones, fibres nerveuses ou musculaires;
- des dentrites ou entrées des neurones;
- les axones, ou sorties du neurone vers d'autres neurones ou fibres musculaires;
- le noyau qui active les sorties en fonction des stimulations en entrée.

Par analogie, le neurone formel est un modèle appelé modèle de perceptron qui se caractérise par un état interne $s \in S$, des signaux d'entrée $x_1, \ldots, x_p$ et une fonction d'activation (Fig.II.2)

$$S = h(x_1 \ldots . x_p) = g(\alpha_0 + \sum_{j=1}^{p} \alpha_j x_j) = g(\alpha_0 + \alpha' x) \qquad \text{(III.1)}$$

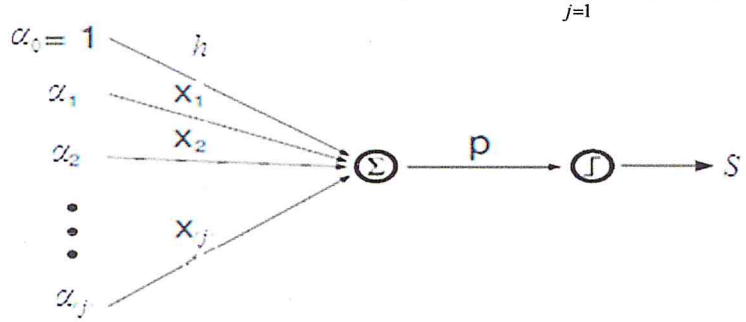
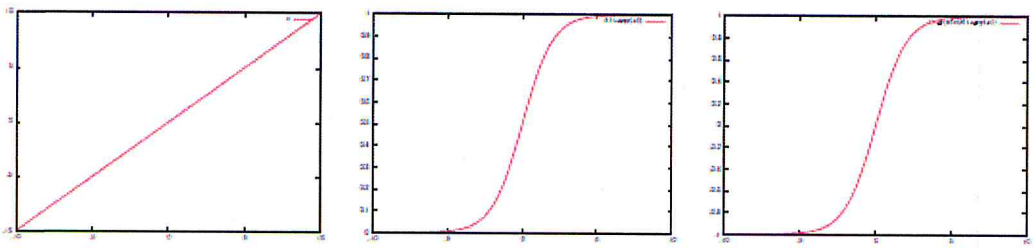


Fig.III.2 Modèle du perceptron.

La fonction d'activation (III.1) opère une transformation d'une combinaison affine des signaux d'entrée, α_0, terme constant, étant appelé le biais du neurone. Cette combinaison affine est déterminée par un vecteur de poids $[\alpha_0, \ldots, \alpha_p]$ associé à chaque neurone et dont les valeurs sont estimées dans la phase d'apprentissage. Ils constituent la mémoire ou connaissance répartie du réseau. Les différents types de neurones se distinguent par la nature g de leur fonction d'activation. Les principaux types sont (Fig.III.3):

- Linéaire : $y = \theta(p) = p$
- Sigmoïde : $g(x) = \dfrac{1}{a + e^{x}}$
- ReLU : $\max(0, x)$
- Tangente hyperbolique : $y = \theta(p) = \dfrac{(1 - e^{x})}{(1 + e^{x})}$



(a) linéaire (b) sigmoïde (c) tangente hyperbolique

Fig.III.3 Fonctions d'activations classiques [53].

La figure III.3 montre les trois fonctions d'activation classiques. Il faut remarquer que la fonction linéaire est dans]-∞;+∞ [, la fonction sigmoïde dans]*0*;*1*[et la fonction tangente hyperbolique dans]-*1*;*1*[.Les modèles linéaires, sigmoïdaux, ReLU, sont bien adaptés aux algorithmes d'apprentissage impliquant une rétro-propagation du gradient car leur fonction d'activation est différentiable; ce sont les plus utilisés. Le modèle à seuil est sans doute plus conforme à la réalité biologique mais pose des problèmes d'apprentissage. Enfin le modèle stochastique est utilisé pour des problèmes d'optimisation globale de fonctions perturbées ou encore pour les analogies avec les systèmes de particules (machine de Bolzman).

III.2.2 Les perceptrons multicouches

Les perceptrons multicouches (MLP) sont capables de traiter des données qui ne sont pas linéairement séparables. Avec l'arrivée des algorithmes de rétro propagation [54]. ils deviennent le type de réseaux de neurones le plus utilisé. Les MLP sont généralement organisés en trois couches, la couche d'entrée, la couche intermédiaire (dite couche cachée) et la couche de sortie. La Fig.III.4 illustre la structure d'un MLP présentant quatre neurones en entrée, trois neurones sur la couche cachée et deux en sortie [55]. Lorsque tous les neurones d'une couche sont connectés aux neurones de la couche suivante, on parle alors de couches complètement connectées [54].

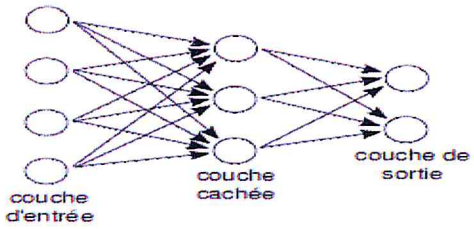


Fig.III.4 Structure d'un perceptron multicouche.

Le passage d'une couche à l'autre peut être formalisé sous forme matricielle. Soit un MLP dont le nombre de neurones sur la couche d'entrée est n_0, n_1 sur la couche cachée et n_2 sur la couche de sortie.

Les perceptrons multicouches ou MLP pour « Multi Layer Perceptron » ont montré leur efficacité comme technique d'apprentissage pour la classification de données. Ils sont en effet capables d'approximer des fonctions non – linéaires complexes afin de traiter des données de grande dimension. Deux approches sont possibles [54]:

- Extraire des caractéristiques directement des données. Classiquement, ces caractéristiques sont extraites par un algorithme choisi par l'utilisateur. Les vecteurs de caractéristiques obtenus sont ensuite présentés en entrée d'un réseau de neurones.
- Présenter l'image en entrée d'un réseau de neurones. L'image nécessite cependant d'être vectorisée, c'est à dire mise sous forme d'un vecteur dont la dimension est égale au nombre de pixels de l'image.

Dans le premier cas, le réseau se contente d'effectuer une classification des vecteurs de caractéristiques. L'extraction des caractéristiques est laissée à la discrétion de l'utilisateur, et le choix de l'algorithme permettant l'extraction des caractéristiques est crucial.

Dans le deuxième cas, plusieurs problèmes se posent :

- Classiquement, les couches d'un réseau de neurones sont complètement connectées, c'est à dire que la valeur d'un neurone d'une couche n va dépendre des valeurs de tous les neurones de la couche $(n-1)$. Ainsi le nombre de connexions (et donc de poids, de paramètres) peut être très grand.
- L'inconvénient majeur des MLP appliqués à des images est qu'ils sont peu ou pas invariants à des transformations de l'entrée, ce qui arrive très souvent avec des images (légères translations, rotations ou distorsions).
- Enfin, les MLP ne prennent pas en compte la corrélation entre pixels d'une image, ce qui est un élément très important pour la reconnaissance des formes.

III.3 Les réseaux de neurones convolutionnels ou CNN

Les réseaux de neurones convolutionnels ou CNN pour « Convolutional Neural Network » sont une extension des MLP permettant de répondre efficacement aux principaux défauts des MLP. Ils sont conçus pour extraire automatiquement les caractéristiques des images d'entrée, sont invariants à de légères distorsions de l'image, et implémentent la notion de partage des poids permettant de réduire considérablement le nombre de paramètres du réseau. Ce partage des poids permet en outre de prendre en compte de manière forte les corrélations locales contenues dans une image. Les CNN ont initialement été inspirés par la découverte faite par Hubel et Wiesel [56] de neurones sensibles aux aspects locaux et sélectifs en orientation. Les modèles du CNN sont bâtis sur le même modèle que les perceptrons multicouches précédemment décrits. Cependant, il convient de souligner que les différentes couches intermédiaires sont plus nombreuses. Chacune des couches intermédiaires va être subdivisée en sous partie, traitant un sous problème, plus simple et fournissant le résultat à la couche suivante, et ainsi de suite [57].

III .3.1 Le Concept général d'un CNN

L'architecture classique d'un CNN est une suite de couches de convolution intercalées par des couches de sous-échantillonnage (**Pooling** en anglais). Les résultats d'une couche de pooling sont suivis par une fonction d'activation qui est une couche de correction, souvent appelée par abus 'ReLU' abréviation de (Unités Rectifié linéaires). Cette dernière sert par la suite comme entrées à la couche de convolution suivante. L'architecture globale d'un CNN est illustrée sur la Fig.III.5 [58].

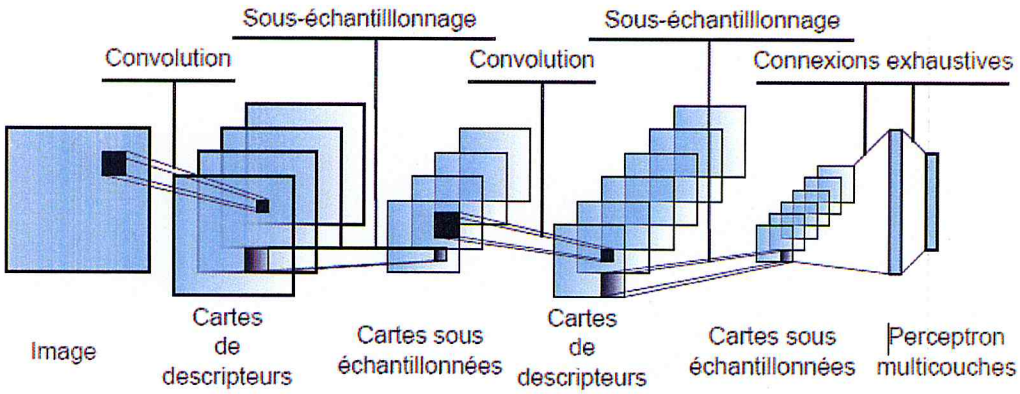


Fig. III.5 L'architecture d'un réseau convolutionel.

Les premières couches servent à extraire des descripteurs efficaces, tandis que les couches finales appelées aussi couches entièrement connectées (FC) sont interconnectées de la même manière qu'un perceptron multicouches. Elles sont utilisées pour la classification. Les convolutions permettent de conserver les dépendances locales, tandis que le sous-échantillonnage permet de compresser l'information en réduisant la taille de l'image intermédiaire.

De tels réseaux sont conçus afin d'apprendre des descripteurs efficaces en même temps que les classifieurs. Cependant, les approches récentes tendent à montrer que l'utilisation des descripteurs combinés avec des classifieurs donnent de bons résultats [59]. Les paramètres à apprendre sont les noyaux de convolution ainsi que les poids entre les couches. Le nombre de noyaux, le nombre de couches, le nombre de neurones sont autant de paramètres à ajuster. La règle d'apprentissage est la retro-propagation de l'erreur [60].

La sortie d'une couche de convolution est appelée carte de descripteurs (featuremap en anglais). Les filtres appris lors de cette étape permettent de capturer les dépendances locales des pixels. La k^{iéme} carte $M_k(I)$ calculée par le noyau de convolution W_k sur l'entrée I est donnée par :

$$M_k = h(I * W_k + b_k) \quad \text{(III.2)}$$

Avec h une fonction d'activation classique comme la sigmoïde ou la tangente hyperbolique, b_k est un biais, W_k et b_k sont appris par retro-propagation de l'erreur.

La propagation de l'erreur s'effectue ainsi :

$$E_k = W * E_{k+1} \quad \text{(III.3)}$$

III.3.1.1 Couche de convolution

La pièce maîtresse d'un CNN est la couche convolutive (CONV) [51]. Elle est constituée de plusieurs filtres (ou noyaux) de convolution à appliquer sur une matrice d'entrée.La couche de convolution est le bloc de construction de base d'un CNN. Trois paramètres permettent de dimensionner le volume de la couche de convolution **la profondeur, le pas et la marge :**

1. **La profondeur de la couche :** nombre de noyaux de convolution (ou nombre de neurones associés à un même champ récepteur).
2. **Le pas (stride)** contrôle le chevauchement des champs récepteurs. Plus le pas est petit, plus les champs récepteurs se chevauchent et plus le volume de sortie sera grand.
3. **La marge (à 0) ou zero padding :** parfois, il est commode de mettre des zéros à la frontière du volume d'entrée. La taille de ce zero-padding est le troisième hyperparamètre. Cette marge permet de contrôler la dimension spatiale du volume de sortie. En particulier, il est parfois souhaitable de conserver la même surface que celle du volume d'entrée.

La taille spatiale du volume de sortie peut être calculée en fonction de la taille du volume d'entrée *Wi*, la taille du filtre utilisé *K*, le pas *S* et la taille de la marge *P*.Le nombre de neurones du volume de sortie est donné comme suit [51] :

$$W_o = \frac{W_i - K + 2P}{S} + 1 \quad \text{(III.4)}$$

La détermination de la marge dépend du pas utilisé. Sa valeur est donnée par la formule suivante [62] :

$$P = \frac{(K-1)}{2}$$

III .3.1.2 Couche de pooling (POOL)

Le pooling réduit la taille spatiale d'une image intermédiaire, réduisant ainsi la quantité de paramètres et de calcul dans le réseau. Il est donc fréquent d'insérer périodiquement une couche de pooling entre deux couches convolutives successives d'une architecture CNN pour contrôler le sur-apprentissage.

La forme la plus courante est le « Max pooling » illustrée par la Fig.III.6 [59].

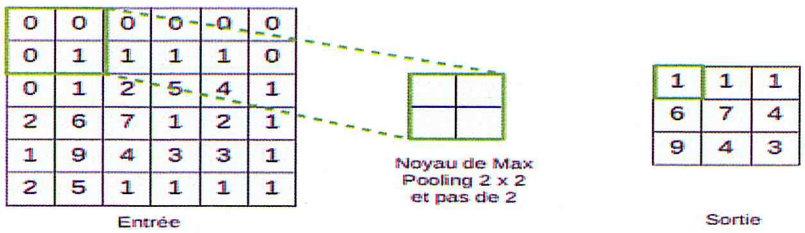


Fig.III.6 Illustration du Max Pooling.

III.3.1.3 Fonction d'activation (ReLU)

La fonction ReLU permet de garder les valeurs positives et attribue la valeur zéro aux valeurs négatives. La plupart des réseaux neuronaux actuels utilisent $F(x) = \max(0, x)$ comme ReLU[62].

III.3.1.4 Normalisation

L'opération de normalisation est appliquée pour chaque couche. Il existe différentes approches de normalisation telles que la couche de canal-sage normalisation [62], qui normalise le vecteur à chaque espace emplacement dans la carte d'entrée, soit dans la même entité carte ou à travers les cartes successives, en utilisant la norme L1 ou L2.

III.3.1.5 Couche entièrement connectée

Après plusieurs couches de convolution et de max-pooling, le raisonnement de haut niveau dans le réseau neuronal se fait via des couches entièrement connectées (FC). Les neurones dans FC ont des connexions vers toutes les sorties de la couche précédente. Leurs fonctions d'activation peuvent être calculées avec une multiplication matricielle suivie d'un décalage de polarisation.

III.3.1.6 Couche de perte

La couche de perte (LOSS) spécifie comment l'entrainement du réseau pénalise l'écart entre le signal prévu et réel. Elle est normalement la dernière couche dans le réseau. Diverses fonctions de perte adaptées à différentes tâches peuvent y être utilisées. La fonction « Softmax» permet de calculer la distribution de probabilités sur les classes de sortie.

III.4Choix des paramètres

Les CNNs utilisent plus de paramètres qu'un MLP standard. Même si les règles habituelles pour les taux d'apprentissage et des constantes de régularisation s'appliquent toujours, il faut prendre en considération les notions de nombre de filtres, leur forme et la forme du max pooling [62].

III.4.1 Nombre de filtres

La taille des images intermédiaires diminue avec la profondeur du traitement, les couches proches de l'entrée ont tendance à avoir moins de filtres, tandis que les couches plus proches de la sortie peuvent en avoir davantage. Pour égaliser le calcul à chaque couche, le produit du nombre de caractéristiques et le nombre de pixels traités est généralement choisi pour être à peu près constant à travers les couches.

Pour préserver l'information en entrée, il faudrait maintenir le nombre de sorties intermédiaires (nombre d'images intermédiaire multiplié par le nombre de positions de pixel) pour être croissante (au sens large) d'une couche à l'autre. Le nombre d'images intermédiaires contrôle directement la puissance du système, et dépend du nombre d'exemples disponibles ainsi que de la complexité du traitement.

III.4.2 Forme du filtre

Les formes de filtre varient grandement dans la littérature. Ils sont généralement choisis en fonction de l'ensemble de données. Le défi est donc de trouver le bon niveau de granularité de manière à créer des abstractions à l'échelle appropriée et adaptée à chaque cas.

III.4.3 Forme du Pooling

La forme la plus courante est le « Max pooling » dont les valeurs typiques sont 2x2. De très grands volumes d'entrée peuvent justifier un pooling 4x4 dans les premières couches. Cependant, le choix de formes plus grandes va considérablement réduire la dimension du signal, et peut entraîner la perte de trop d'information. Pour ne pas tomber dans le problème de sur apprentissage, des méthodes de régularisation sont appliquées telles que :

- **Dropout :** Le Dropout [62] est une technique où des neurones sélectionnés au hasard sont ignorés (temporairement) pendant l'apprentissage (Fig. III.7). Cela signifie que leur contribution à l'activation des neurones qui leur succèdent est temporairement supprimée lors de la phase de propagation et toutes les mises à jour de poids ne sont pas appliquées au neurone lors de la phase de retro-propagation. Lorsque des neurones sont supprimés au hasard du réseau pendant l'apprentissage, les autres neurones devront intervenir et gérer la représentation requise pour faire des prédictions pour les neurones manquants. Lors de la phase d'apprentissage, pour chaque itération, un neurone est gardé avec une probabilité p, sinon il est supprimé.

- **DropConnect :** Le DropConnect est une évolution du dropout, où on ne va non plus éteindre un neurone, mais une connexion (l'équivalent de la synapse), et ce, de manière toujours aléatoire. Les résultats sont similaires (rapidité, capacité de généralisation de l'apprentissage), mais présentent une différence au niveau de l'évolution des poids des connexions. Une couche FC avec un DropConnect peut s'apparenter à une couche à connexion "diffuse".

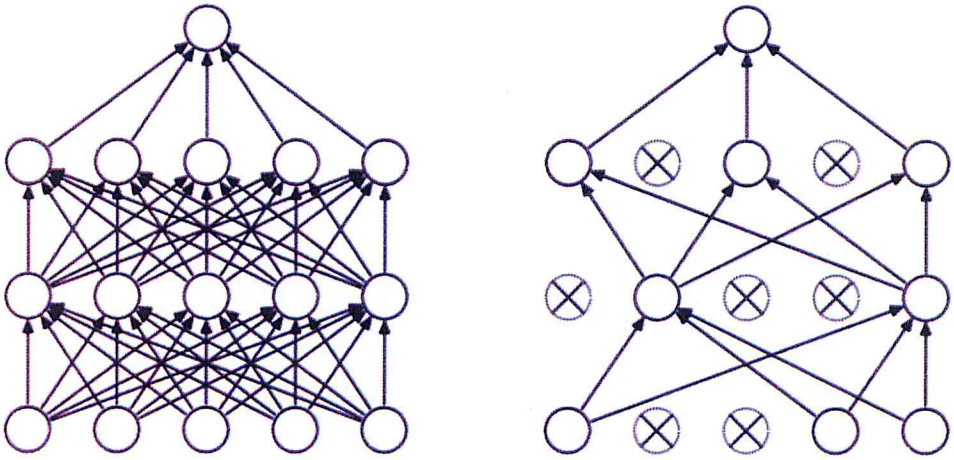


Fig. III.7 Réseau avant et après l'opération du dropout.

III.5 Les architectures neuronales classiques

Nous présentons dans ce paragraphe les architectures de réseaux convolutifs profonds utilisées couramment.

Les architectures proposées dans la littérature ont une forte tendance à devenir de plus en plus profonde avec les années. Autrement dit, il semble que, plus le réseau est profond, plus les performances sont bonnes. Néanmoins, cette profondeur implique de faire face à certaines difficultés notamment en termes de temps de calcul et d'optimisation durant l'apprentissage. C'est pourquoi la communauté reste très active sur la problématique de conception d'architectures neuronales. Des architectures standards sont abondamment utilisées en vision pour deux raisons principales.

La première raison est qu'elles permettent d'inter-comparer facilement les méthodes basées sur les CNN. En d'autres termes, bien que certains travaux se focalisent sur l'étude des architectures neuronales, la majorité des méthodes de vision réutilisent des CNN déjà appris et les modifient pour concevoir de nouvelles architectures répondant à des tâches particulières.

La seconde raison, est en lien avec la difficulté d'apprendre les réseaux profonds du fait de leur grand nombre de paramètres et du manque de données d'apprentissage.
Parmi Les architecture CNN, nous citons les plus connues et les plus répandues.

III.5.1 AlexNet

L'architecture AlexNet a été proposée par *Krizhevskyet et al.[*63]. Cette architecture utilise cinq couches de convolution et trois couches de pooling. La taille des noyaux de convolution est variable (11×11, 5×5, 3×3) en fonction de la couche considérée. La fonction d'activation utilisée entre chaque couche est la fonction ReLU. Après le passage de l'image dans les couches de convolution (CONV), de pooling (MaX-POOL) et d'activation, une carte de caractéristiques est obtenue. Celle-ci est envoyée dans un perceptron multicouche (MLP) composé de deux couches cachées et d'une couche de sortie Fig. III.8.

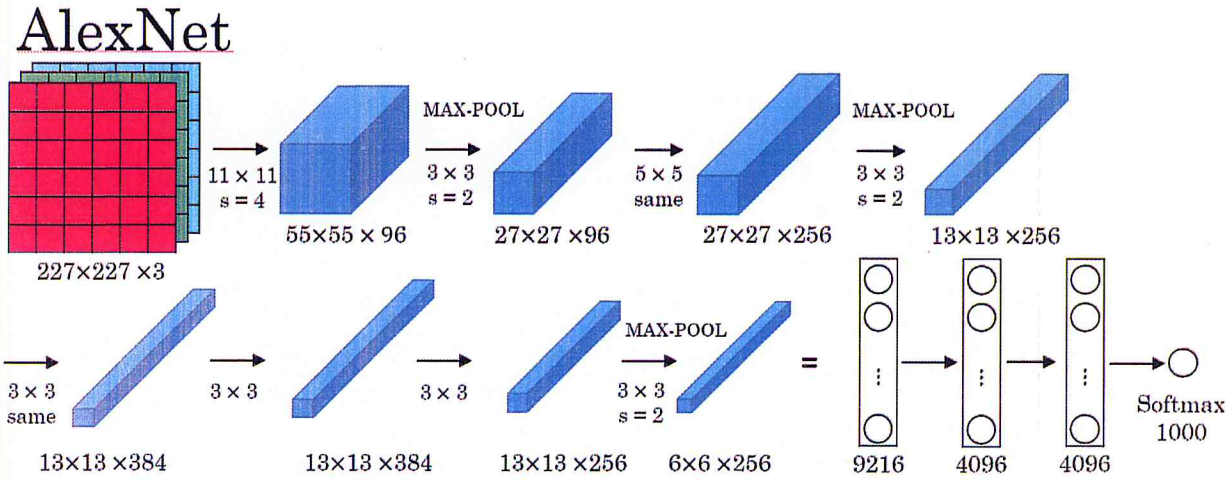


Fig. III.8 Architecture de AlexNet.

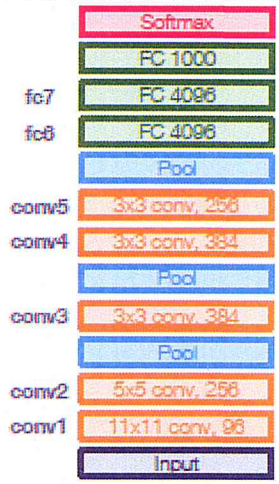
Les entrées à l'architecture AlexNetsont des images de taille 227x 227x3. La valeur 3 correspond au nombre de plan dans l'espace de couleur RVB. La taille des images à l'entrée de cette couche est réduite à la taille 55x55 selon l'équation (III.4), puis 96 filtres de taille 11x11 avec un pas *s* égal à 4 sont appliqués. Ensuite une couche MAX-POOL de taille 3x3 et un pas s de 2 sont appliqués. Les résultats obtenus à la sortie de la première couche sont des images de 27x27.

Sur ces dernières 96 filtres de taille 5x5 avec le même *s*. Puis à nouveau un MAX-POOL de taille 3x3 et *s* égale à 2 sont appliqués jusqu'à la taille des images 13x13 où 256 filtres sont appliqués

Le résultat à la sortie est un vecteur de 9216 neurones qui est à son tour connecté à deux autres couches FC de 4096 nœuds pour donner en sortie un vecteur caractéristique de 1000 classes.

Le tableau III.1 résume les différents paramètres de l'architecture AlexNet.

Tableau III. 1 Paramètres de l'architecture AlexNet.



II.5.2 ResNet

L'architecture ResNet a été introduite par Zhang et al.**[65].** Cette architecture permet l'apprentissage de réseaux très profonds (plus de 150 couches). La difficulté à apprendre des réseaux aussi profonds est notamment liée à la rétro-propagation du gradient. Plus le réseau est profond, plus le gradient est faible pour la mise à jour des poids des couches de plus bas niveau (les premières couches). L'idée développée dans ResNet est l'utilisation de connexions résiduelles permettant une meilleure optimisation des réseaux très profonds. Une connexion résiduelle permet de passer l'entrée dans deux filtres de convolution mais également de passer directement cette entrée aux couches suivantes Fig. III.9 [65].

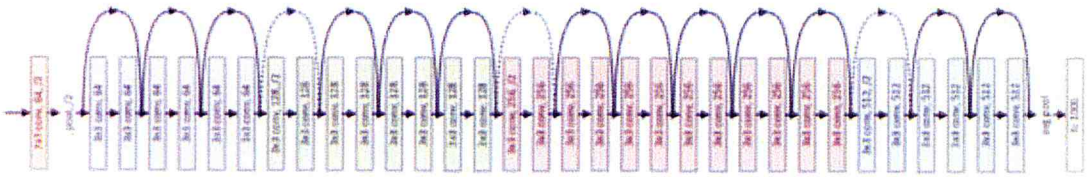


Fig.III.9 Architecture du ResNet (34 couches).

III.5.3 VGGNet

VGGNet. a été introduit par *Simonyanet et al.* en 2014 [66]. Le schéma bloc de VGGNet est représenté sur la figure III.11. Au lieu d'utiliser une seule convolution par niveau de profondeur comme AlexNet, cette architecture utilise 16 séquences de convolution dont les filtres sont de taille plus petite que ceux utilisés dans AlexNet (noyau de taille 3×3) Fig. III.10 [57].

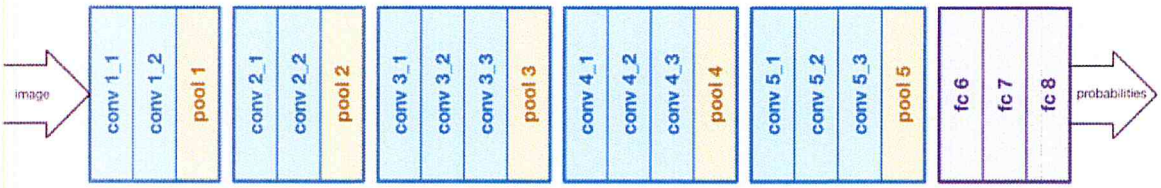


Fig. III.10. Architecture du VGGNet.

III.5.4 GoogleNet.

Cette architecture neuronale convolutive a été créée par *Szegedy et al.* en 2015 [68]. Elle permet une réduction du temps de calcul par rapport à l'architecture VGG présentée précédemment. Pour cela, le GoogLeNet est composé de plusieurs couches appelées modules d'inception. Chaque module est composé de plusieurs couches de convolution de taille 1x1, 3x3 et 5x5, exécutés en parallèle sur la carte de caractéristiques résultant de la couche précédente. Des filtres additionnels permettent de réduire la dimension des cartes de caractéristiques ce qui permet un gain important de temps de calcul.

La Figure III.11 illustre une couche d'inception et la Figure III.12 représente l'architecture globale du GoogLeNet. D'autres modules d'inception ont par la suite été proposés notamment Inception V2 et V3.

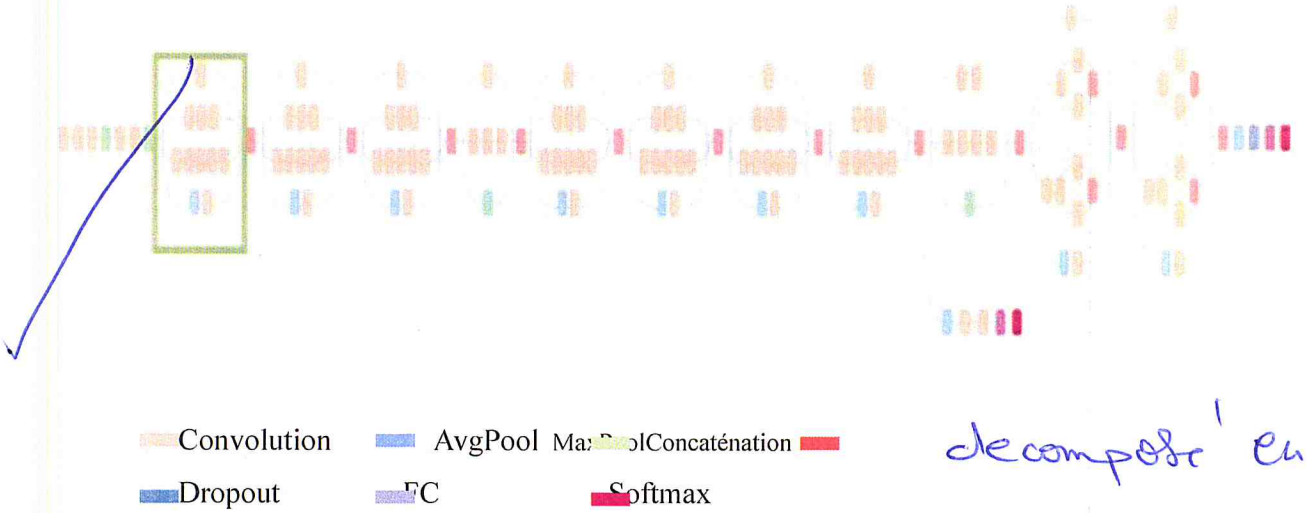


Fig. III.11 Architecture GoogleNet.

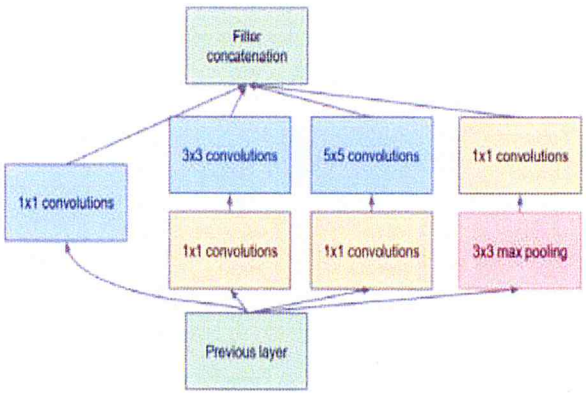


Fig.III.12Module d'inception.

III.6Conclusion

Dans ce chapitre, nous avons introduit les réseaux de neurones et plus particulièrement les réseaux neuronaux convolutifs profonds (CNN) utilisés tout au long du travail présenté dans ce mémoire. Ces modèles permettent d'apprendre, grâce à l'apprentissage supervisé, les caractéristiques visuelles discriminantes à extraire sur les images afin de résoudre des problématiques de détection et de classification. Parmi les architectures citées, nous avons parlé de l'architecture VGGNet qui est utilisée dans la plateforme de détection et d'alignement de visage qui sera détaillée dans le prochain chapitre.

Chapitre IV

Implémentation

IV.1 Introduction

Vu que la détection des visages Humains a pris de l'ampleur pour devenir un domaine de recherche très actif et vu la difficulté de ce domaine à cause de la multitude des paramètres qu'il faut prendre en compte tels que la variation de posture, l'éclairage, le port de lunettes; de barbe; de moustaches, vieillesse; etc.
Ce dernier chapitre est consacré à l'implémentation de la plateforme de détection et l'alignement de visage.

Le développement de la plateforme est en langage de programmation python, avec les bibliothèques OpenCV pour pouvoir prendre des photos à l'aide de la webcam d'un ordinateur (pour la détection et l'alignement en temps réel), et les photos stocké dans le path de l'application.
Pour la détection de visage il utilisera la méthode classique dite HOG (Histogramme de gradient orienté) que nous détaillerons plus tard, et pour l'alignement il utilisera les CNN que nous également détaillerons plus tard.

Ce chapitre est divisée en deux parties dans la première partie nous décrirons la méthode développée ainsi que l'architecture les modèles de détection et d'alignement utilisés. La deuxième partie portera sur la description des logiciels utilisés pour l'implémentation et la mise en œuvre du programme de détection et alignement du visage.

IV.2 Les méthodes développées

IV.2.1 Implémentation méthode 1 (viola et jones)

Nous avons choisi la méthode de Viola-Jones car elle est largement utilisée dans la détection des objets particulièrement le visage. Comme il a déjà été mentionné dans le premier chapitre, la détection de visage a beaucoup d'applications dans la vie quotidienne. C'est une première étape dans n'importe quels systèmes de traitement de visage tels que reconnaissance de visage, vérification de visage. Avec la bibliothèque OpenCV, il est assez facile de détecter un visage de face dans une image en utilisant le détecteur de visage Haar Cascade (connu comme la méthode de Viola-Jones). Etant donné un fichier ou une vidéo en direct, le détecteur de visage examine chaque emplacement de l'image et classifie comme visage ou non visage. Le classifieur utilise des données stockées dans un fichier XML pour décider comment classifier chaque localisation image. OpenCV est livré avec plusieurs classifieurs différents pour la détection de visage dans des poses frontales, ainsi que la détection de certains visages de profil, la détection des yeux, la détection des corps...etc. Nous pouvons utiliser la fonction Cv2.cascadeClassifier avec l'un de ces autres détecteurs, mais le détecteur des visages de face est le seul qui est très fiable. Pour la détection des données XML,

Nous pouvons choisir l'un de ces classifieurs Haar Cascade d'OpenCV (dans le répertoire "opencv/data/haarcascade"):

- haarcascade_profileface.xml
- haarcascade_frontalface_alt.xml

Nous avons choisi le fichier ***haarcascade_frontalface_alt.xml*** pour l'application car il correspond parfaitement à notre objectif.

On ouvre le flux caméra avec ***l'instruction vid.read() : frame= vid.read()***

La détection de visage sur l'image capturée est obtenue en utilisant la procédure de Haar Cascade : **.face_cascade.detectMultiScale(gray, 1.3, 1,minSize=(5, 5))** où minSize(5,5) représente la taille minimale du rectangle du visage détecté. Le résultat obtenu est un rectangle de visage pour la région détectée dans l'image donnée.

La Fonction **face_cascade.detectMultiScale(gray, 1.3, 1,minSize=(5, 5))** utilise en paramètre l'image récupérée pendant la boucle de traitement.. C'est sur cette image que toutes les opérations sont effectuées par la suite. Dans OpenCV, la détection de visage par la méthode de Viola-Jones est déjà implémentée dans la fonction cvHaarDetectObjects. Le résultat de cette fonction est une série d'objets d'un même type qui ont subi les critères de sélection définis par le classifieur. Dans notre cas les différents visages sont détectés (Fig. IV.1).

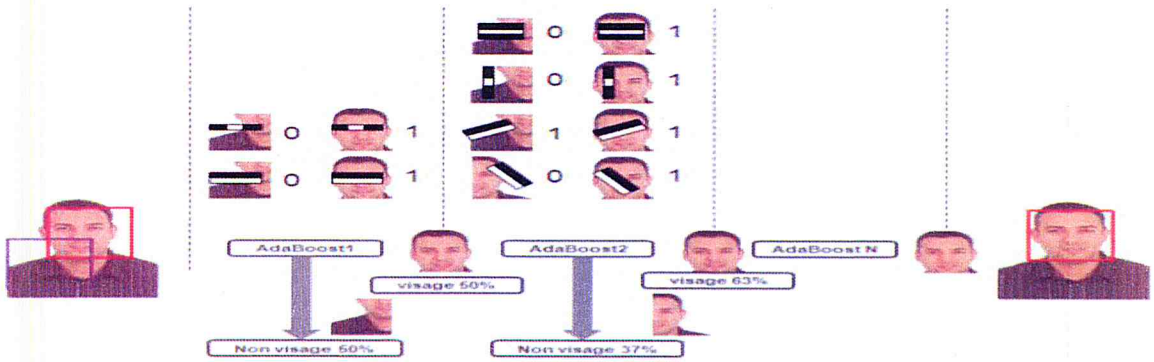


Fig. IV.1 Un exemple de visage détecté avec la méthode de Viola-Jones.

IV.2.2 Implémentation méthode 2 (Avec CNN architecture Mini-VGGNet)

Rappelons pour commencer l'objectif à atteindre. On possède une image et on veut détecter un visage et les objets caractéristiques du visage.

La plateforme développée comprend deux étapes la détection des visages et l'alignement et le raffinement. La figure IV.2 illustre le schéma blocs de la plateforme. Le système est divisé en deux sous-systèmes. Un sous-système pour la localisation des repères faciaux pour les composants du visage yeux, nez, bouche, et sourcils, et le second est destiné pour la localisation des points de repères au niveau du contour du visage (menton).

Pour localiser le visage, nous avons employé la méthode de détection de visage classique HOG (Histograms of Oriented Gradients) [69] et [70]. Classique basée sur l'Histogramme de HOG appartenant à la bibliothèque Dlib.

Pour le processus d'alignement des points de repère du visage, nous avons utilisé l'architecture Mini VGGNET (VGG 7).

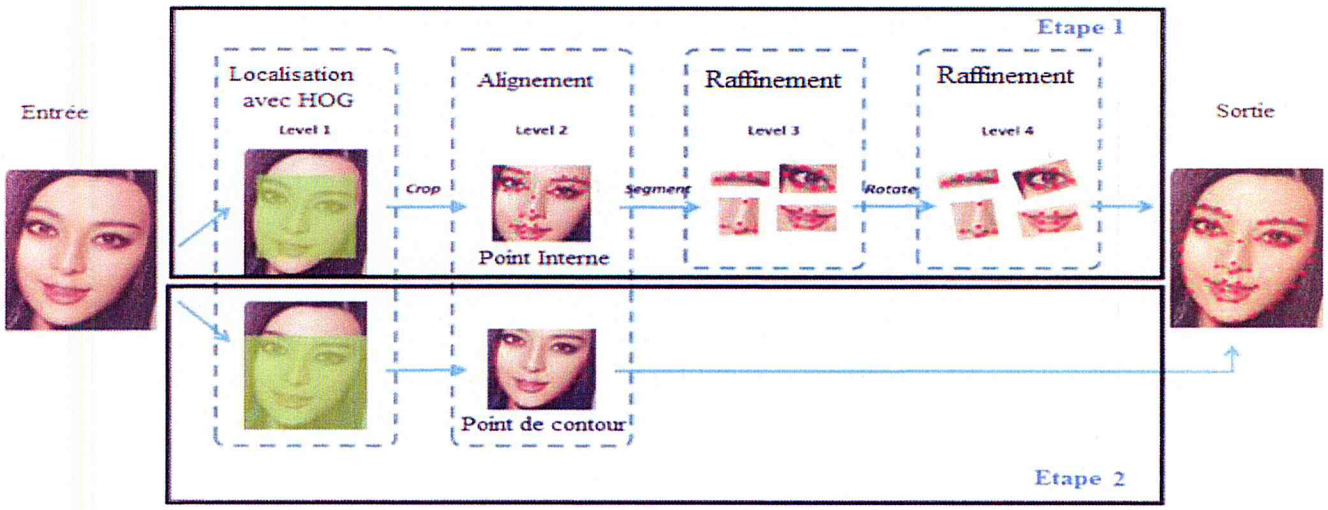


Fig.IV.2 Synoptique de la plateforme développée

IV.2.2.1 Localisation du visage à partir de la bibliothèque Dlib

Pour localiser les visages, nous avons utilisé la fonction implémentée par Dlib basée sur la combinaison de HOG et le classifieur linéaire de machines à vecteurs de supports SVM (Support Vector Machine) :

detector = dlib.get_frontal_face_detector().

Cette fonction examine chaque emplacement de l'image et la classifie comme visage ou non visage. Le classifieur utilise des données stockées dans un fichier XML pour décider de la classification de l'image.
Les descripteurs HOG sont introduits par Dalal et Triggs. L'idée essentielle derrière l'histogramme de gradient orienté c'est que l'apparence locale et la forme d'objet dans une image peut être décrite par la distribution d'intensité des gradients ou de direction des contours.
La mise en œuvre de ces descripteurs est obtenue en divisant l'image en petites régions connectées, appelées cellules. Pour chaque cellule un histogramme des directions de gradient ou des orientations de contour est calculé. L'histogramme est ensuite comparé avec des histogrammes de visages prédéfinis afin de détecter si un visage est présent dans l'image en entrée. Ensuite le classifieur SVM mesure l'apport des HOG et prédit le pourcentage de détection de visage.

IV.2.2.2Alignement du visage à l'aide des CNN

La méthode utilisée dans Dlib pour la détection des points de repère faciaux est une implémentation du papier de Erjin Zhou et al. (2013): "Extensive Facial Landmark Localisation withcoarse to-fine Convolutional Network Cascade" [71]. Cette méthode consiste en l'apprentissage d'une cascade de régresseurs à partir d'un jeu de données. Pour les repères faciaux, nous utilisons le terme points internes pour dénoter les 51 points faciaux générer au niveau du premier sous-système (Fig. IV3(a)), et les points de contour pour les 17 points localisés sur le contour (second sous-système) (Fig. IV3(b)). En tout 68 points faciaux caractérise un visage.

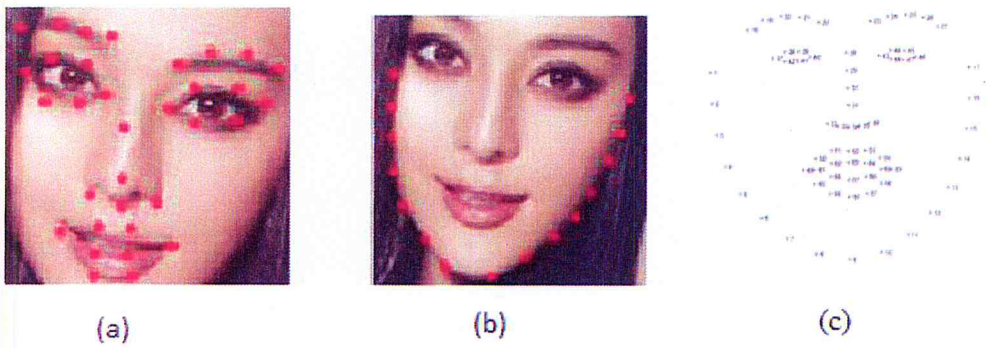


Fig. IV.3 (a) points internes ; (b) points de contour (c) Marquages des 68 points utilisés pour les annotations

La stratégie "**diviser pour mieux régner**" est adoptée, elle consiste à divise la tâche en deux étapes: d'abord pour trouver l'ensemble de position des points de repères de contour, puis calculer la position des points de repères internes relative à l'intérieur de région. De cette façon, le fardeau est partagé entre les réseaux dans différents niveaux, et de bonnes performances sont atteintes par les réseaux de taille modérée.

Nous avons utilisé les CNN comme bloc de construction de base du système. Le réseau prend les pixels bruts en entrée et effectue une régression sur les coordonnées des points souhaités. L'architecture du modèle adopté est présentée sur la Fig.IV.4. Le modèle est composé de cinq couches de convolution © et deux couches entièrement connectées (FC).

L'image en entrée est de taille 224x224, l'image passe d'abord à la première couche de convolution. Cette couche est composée de 96 filtres de taille 7x7, avec un pas (stride) de 2. Chaque couche © est suivie par une fonction d'activation Relu. Cette fonction force les neurones à retourner des valeurs positives, suivi d'un Maxpooling (MP) de 3x3 pour réduire la taille de l'image ainsi que la quantité de calcul. Le résultat issue de cette derniere est 96 cartes de caractéristiques de taille 112x112. Les 96 cartes de caractéristiques obtenues sont les entrées de la deuxième © qui est composée aussi de 256 filtres de taille 5x5 suivie par le Relu et une couche MP de taille 2x2. À la sortie de cette couche, nous aurons 256 cartes de caractéristiques de taille 56x56. La même opération se répète pour la troisième, la quatrième et la cinquième ©. Elles sont composées de 512 filtres de taille 3x3, d'une fonction d'activation Relu et une couche MP de taille 3x3. A la sortie de la cinquième couche © , nous aurons 512 cartes de caractéristiques de taille 7x7. Les cinq couches (C1, C2, C3, C4, C5) sont suivies par deux couches FC. La première couche a 4096 neurones où la fonction d'activation utilisée est le Relu, et la deuxième couche est de 1000 neurones suivie par la fonction d'activation softmax. Cette dernière permet de calculer la distribution de probabilité et de déduire le résultat.

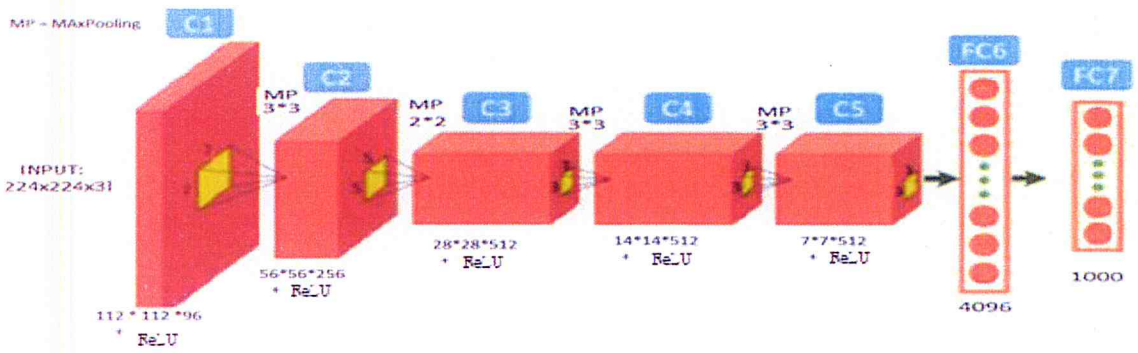


Fig.IV.4 Architecture du modèle utilisé (Mini-VGGNet)

Tableau IV.1 Résumé de l'architecture du modèle utilisé (Mini-VGGNet)

La convolution	cartes de caractéristiques	Taille	Taille Filtre	Fonction d'activation	MaxPooling
La couche Conv1	96	112x112	7x7	Relu	3x3
La Couche Conv2	256	56x56	5x5	Relu	2x2
La couche Conv3	512	28x28	3x3	Relu	3x3
La couche Conv4	512	14x14	3x3	Relu	3x3
La couche Conv5	512	7x7	3x3	Relu	//////////////

IV.2.2.3 Le processus de raffinement

Le réseau du premier niveau prédit la localisation pour le visage, le réseau du deuxième niveau prédit une estimation initiale des positions des points internes et prédit les points de contour séparément. Le processus de raffinement est appliqué uniquement sur les points internes du premier sous système. Il est appliqué sur chacun des composants du visage (les yeux, les sourcils, le nez, et la bouche) séparément (Fig. IV.5). Le même modèle de CNN est appliqué pour le raffinement des points de repère.

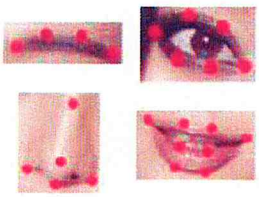


Fig. IV.5 raffinement des points internes

Afin d'améliorer la localisation des points de repère, l'angle de rotation de chaque composant est estimé et corrigé en position verticale, et les patches tournés alimentent (entrée) le réseau de quatrième niveau pour les résultats finaux.

IV.3 Entraînement (Training)

Le modèle de marquage utilisé est prédéfini, nous l'avons importé de la bibliothèque **Caffe.** C'est un modèle développé pendant l'entrainement et est utilisé pour identifier la position 68 points de repère sur les visages humains. L'ensemble de données pour l'entrainement et le développement du modèle est issu des bases de données AFW, AFLW et 300-w(300-W 'interne',300-W 'externe'). Nous fournissons un aperçu des ensembles de données ci-dessus Fig. IV.6.

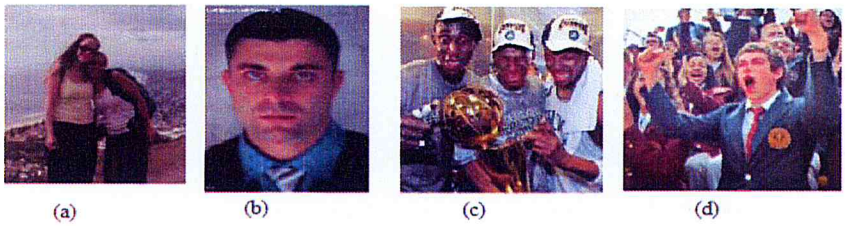


Fig. IV.6. Images annotée des (a) AFW,(b) AFLW,(c) 300-W 'interne' , (d) 300-W 'externe'

AFW: la base de données Annotées (AFW :*Annoted Faces in-the-wild*) contient 250 images avec 468 visages. Six points de repère faciaux pour chaque visage. Fig. IV.5.(a) représente une image annotée d'AFW. Téléchargeable depuis le lien :https://www.ics.uci.edu/~xzhu/face/AFW.zip'

AFLW: Les repères faciaux annotés dans la nature (AFLW : *The Annoted Facial Landmarks in-the-wild*) contient 25 000 images de 24 686 sujets téléchargé. Les images contiennent une large gamme des poses et des occlusions du visage naturel. Les annotations faciales sont disponibles pour l'ensemble de la base de données. Chaque annotation comprend 21 points de repère (Fig. IV.5(b)). Téléchargeable depuis le lien : http://mmlab.ie.cuhk.edu.hk/archive/CNN/data/train.zip'

Les bases de données mentionnées ci-dessus couvrent des sujets différents, poses, illumination, occlusion etc..

Pour tester le réseau implémenté nous utiliserons la base de données 300-W qui vise à examiner la capacité des systèmes pour gérer les visages naturels et sans contrainte d'âge.

L'ensemble des tests doit couvrir différentes variantes telles que pose, expression, illumination, arrière-plan, occlusion et qualité de l'image. De plus, les images de test devraient couvrir de nombreuses expressions.

La base de données 300_w contenant 2x300 images faciales capturées dans monde réel, qui consiste en 300 images intérieures et 300 extérieures en nature. . Il couvre une grande variation d'identité, d'expression, de conditions d'illumination, pose, occlusion et taille du visage.

IV.4 Logiciels et librairies Utilisés dans l'implémentation

L'environnement de travail est important pour le développement des projets spécialement les projets comme projet de fin d'étude pour simplifier, expliciter, et adapter l'application. L'environnement de développement que nous venons d'installer est un ensemble d'outils pour programmeurs conçus pour être utilisés au sein d'un éditeur interactif nommé *Visuel Studio Code,* le langage de programmation utilisé est le python.

IV.4.1 Langage de Programmation : Python

Python est un langage de programmation très polyvalent et modulaire. Il est constitué de différents outils et composants présentés dans FigIV.7.

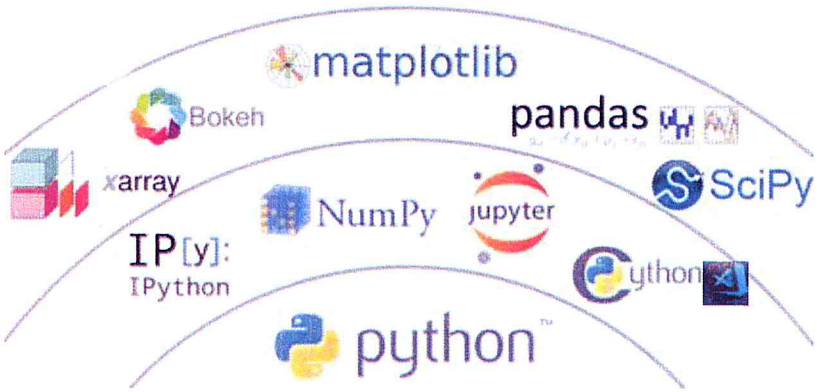


Fig. IV.7 Outil de développement

Pour le calcul scientifique nous utilisons plusieurs bibliothèques comme :

Numpy: Calcul de tableau et calcul matriciel.

Scipy: Outils numériques standards: intégration résolution de système non linéaire.

Matplotlib: Pour tracer des courbes et afficher des résultats scientifiques.

IV.4.2 OpenCV(Open Source Computer vision)

C'est une bibliothèque graphique libre, initialement développée par Intel et maintenant soutenu par la société de robotique Willow Garage, spécialisée dans le traitement d'images en temps réel. Elle est livrée avec une interface de programmation en C, C++, Python et Android. La bibliothèque OpenCV met à disposition de nombreuses fonctionnalités très diversifiées, elle propose la plupart des opérations classiques en traitement d'images telles que ::

- lecture, écriture et affichage d'image et vidéo depuis un fichier ou une caméra ;
- calcul de l'histogramme des niveaux de gris ou d'histogramme de couleur ;
- lissage, filtrage.
- détection d'objets et détection de mouvements, etc….

IV.4. 3 Caffe

Le caffe est un cadre d'apprentissage en profondeur conçu en fonction de l'expression de la rapidité et de la modularité. Il est développé par Berkeley AI Research (BAIR) et par des contributeurs communautaires. Yangqing Jia a créé le projet au cours de son doctorat à l'UC Berkeley. Caffe est publié sous la licence BSD 2-Clause.

IV.4.4 Dlib

Dlib (bibliothèque) est une boîte à outils C ++ contenant des algorithmes d'apprentissage automatique et des outils pour créer des logiciels complexes en C ++ pour résoudre des problèmes du monde réel. Elle est utilisée dans l'industrie et le milieu universitaire dans un large éventail de domaines, y compris la robotique, les dispositifs intégrés, les téléphones mobiles, et les grands environnements informatiques de haute performance.

IV.5 Conclusion

Nous avons présenté dans ce chapitre une approche de classification basée sur les réseaux de neurones convolutionnels, pour cela nous avons utilisé trois modèles avec différents architectures et nous avons montré les différents résultats obtenus en termes de précision et d'erreur. La comparaison des résultats trouvés a montré que le nombre d'époque, la taille de la base et la profondeur de réseaux, sont des facteurs importants pour l'obtention de meilleurs résultats.

Chapitre V

Résultats et Tests

V.1 Introduction

Ce projet est une application de la détection et alignement de visage, il a été développé en utilisant le langage de programmation python.

Dans cette partie, nous avons présenté l'interface de cette application et montrons des résultats des Tests

V.2 Application (NovMDFs):

L'application ***NovMDFs*** offre à l'utilisateur la possibilité de télécharger une image à partir d'un endroit sur son ordinateur, ou à partir d'une caméra pour effectuer la détection et l'alignement des visages en temps réel.

L'application est capable de détecter des visages dans différentes obstacles comme :

- les occultations
- l'éclairage
- les expressions faciales...

V.2.1 Description de l'application

L'interface de notre application contient trois boutons :

Open Image : pour accéder au fichier de PC et télécharger une image ;

Save Image : pour sauvegarder l'image examinée ;

Close : pour fermer l'interface principale.

Pour faciliter la tâche d'exécution de NovMDFs , le menu barre offre une utilité facile exprimer en :

V.2.1.1 Détection de visage en Image

Elle permet de détecter les visages sur des images téléchargées à partir du disque.

- Detect Face Viola-Jones
- Detect Face with VGGNet

V.2.1.2 Détection de visage en Temps réel

NovMDFs permet de détecter le visage en temps réel en utilisant la caméra de notre PC portable dans les différentes méthodes implémentées.

- Detect Face Viola-Jones
- Detect Face with VGGNet

L'interface graphique de l'application NovMDFs que nous avons créé est illustrée dans figure suivante :

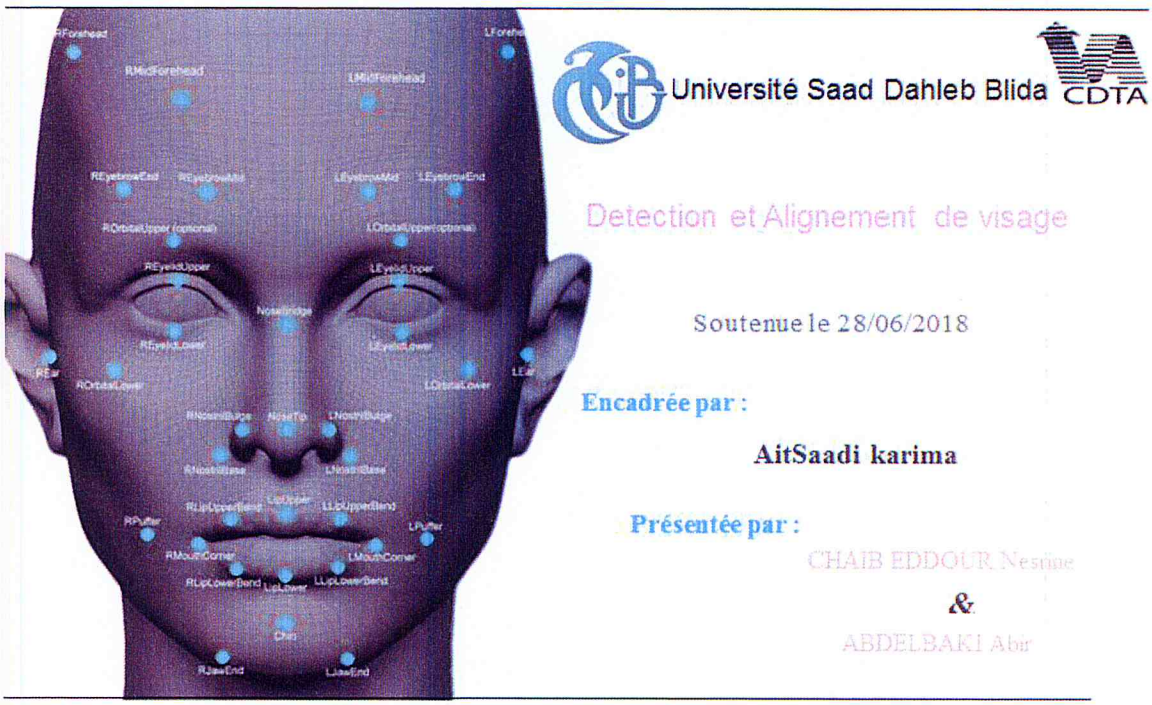


Fig V.1. Interface Principale de NovDFS

V.3 Rappel et précision

Avant d'analyser la qualité de l'extraction, l'image i obtenue après traitement par l'un des algorithmes est comparée avec l'image de vérité terrain correspondante, nous devrons classer les pixels selon les catégories suivantes :

- VP (vrais positifs) : premier plan détecté comme premier plan.
- FP (faux positifs) : fond détecté comme premier plan.
- VN (vrais négatifs) : fond détecté comme fond.
- FN (faux négatifs) : premier plan détecté comme fond.

V.3.1 Rappel et précision

Pour comparer nos résultats obtenus avec la vérité terrain nous calculons le rappel et la précision qui sont deux critères pour mesurer la qualité de l'extraction.

V.3.1.1 Rappel

Le rappel est défini comme le nombre de vrais positifs divisé par le nombre total d'éléments qui appartiennent au premier plan.

$$Rapppel = \frac{TP}{TP + FN} \qquad (V.1)$$

V.3.1.2 Précision

La précision peut être considérée comme une mesure de précision, elle évalue en divisant le nombre d'objets de premier plan correctement détectée par le nombre total de pixels classés en premier plan par l'algorithme.

$$\text{Pr}\,écision = \frac{TP}{TP + FP} \qquad \text{(V.3)}$$

Les valeurs du rappel et de la précision se situent entre 0 et 1. Un bon algorithme qui détecte bien les objets en mouvement a des valeurs de rappel et de précision qui se rapprochent de 1.

Résultat de rappel et précision sur des images :

Dans cette partie nous calculons manuellement le rappel et la précision des méthodes utilisées Viola-Jones, VGGNet (CNN), avec le logiciel (gimp (ubuntu))

Appareil de photo utilisée : Caméra GALAXY S8

Les résultats sont basés sur les valeurs extraites d'un échantillon de 5 images pour chaque méthode Fig. V.2

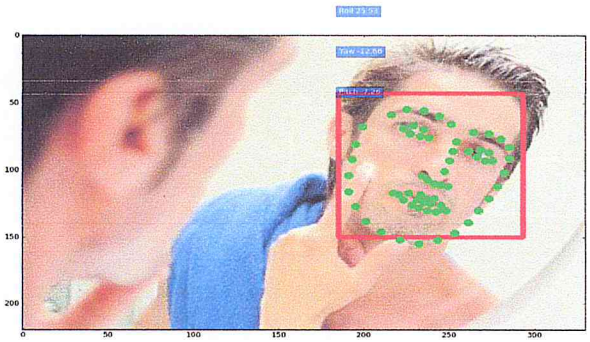


Fig. V.2. Détection et localisation des points de repères faciaux dans une image à partir d'un miroir avec la méthode VGGNet

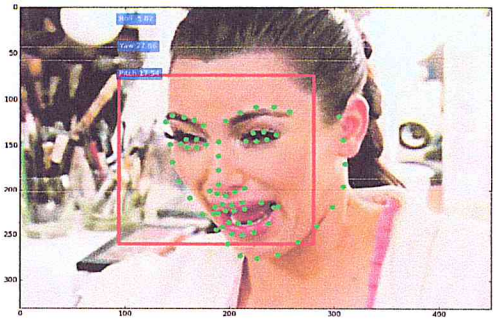


Fig. V.3 Détection et localisation des points de repères faciaux dans une image avec une expression faciale avec la méthode VGGNet

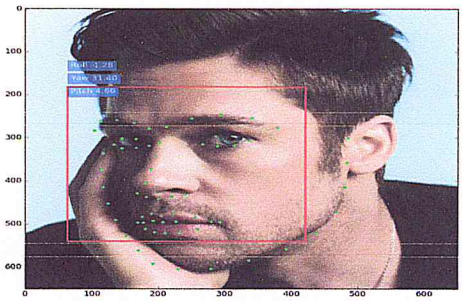


Fig. V.4 Détection et localisation des points de repères faciaux dans une image avec un visage de profile avec la méthode VGGNet

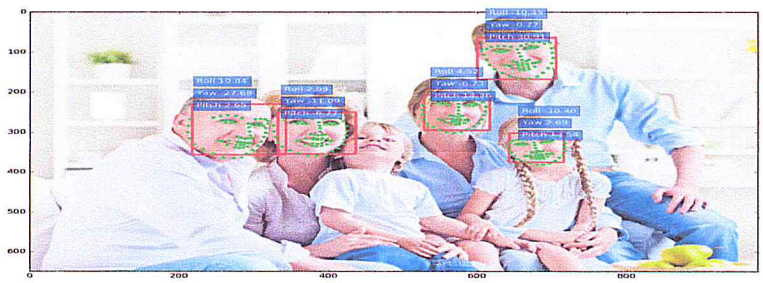


Fig. V.5 Détection et localisation des points de repères faciaux dans une image avec plusieurs de personnes avec la méthode VGGNet

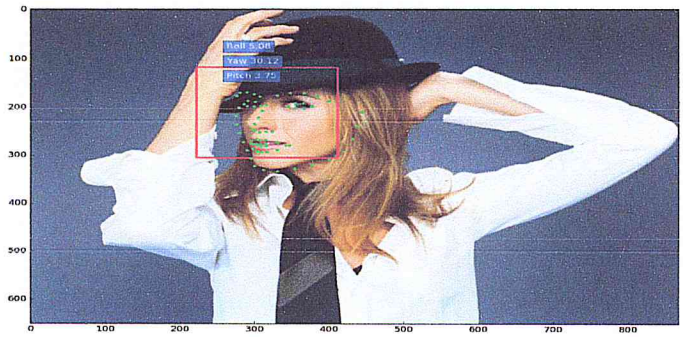


Fig. V.6 Détection et localisation des points de repères faciaux dans une image avec un chapeau à la tête avec la méthode VGGNet

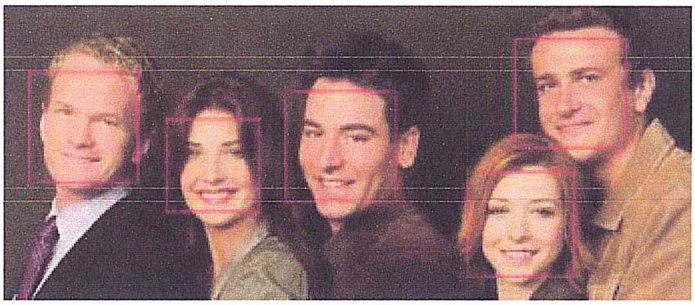


Fig. V.7 Détection de visage dans une image avec plusieurs de personnes avec la méthode de Viola et Jones

Fig. V.8 Détection de visage dans une image avec un chapeau à la tête avec la méthode de Viola et Jones

Ce tableau montre les valeurs de rappel et précision calculées de chaque méthode Table V.1

Table V.1 les valeurs de rappel et précision calculées de chaque méthode

Valeur Méthode	Temps d'exécution	TruePositif	FalsePositif	FalseNegatif	Rappel	précision
Viola-Jones	bien	19800	150	120	0.76	0,91
VGGNet	bien	20400	760	1150	0,96	0,95

D'après les calculs on remarque une nette différence entre les deux méthodes, On voit que le rapport (précision, rappel) de la méthode viola et jones est inférieur au rapport (précision, rappel) de la méthode VGGNet CNN. Et nous avions dit qu'un bon

algorithme qui détecte bien les objets en mouvement a des valeurs de rappel et de précision qui se rapprochent de 1. Donc on constate que la méthode VGGNet est meilleure que la méthode Viola et jones

Conclusion

Dans ce chapitre nous vous présentons les Méthodes utilisées, les tests et les résultats des deux méthodes, Pour obtenir de bons résultats nous avons réalisé plusieurs tests.

D'après les résultats des tests nous avons conclut que la qualité de suivi est meilleures avec la méthode d'alignement VGGN et CNN par rapport à la méthode de viola et jones.

Bibliographie

[1] http://slideplayer.fr/slide/9720594/, consulté, le 24/04/2018.
[2] M.khammari, Détection et suivi de visages en temps Réel sur Flux Vidéo, 2015/2016 these en vue d'optenir du diplôme de doctorant troisieme cycle lmd option traitement dimage et vision artificielle.
[3] T. Sakai, M. Nagao, and T. Kanade, "Computer analysis and classification of photographs of human faces", in Computer Conference pp. 2-7, First USA—Japan January,1972.
[4] M. C. Burl and P. Perona, "Recognition of planar object classes", in IEEE on Computer Vision and Pattern Recognition, pp.223 - 230, San Francisco,June 1996.
[5] M.H. Yang, D. J. Kriegman, and N. Ahuja. "Detecting faces in images: A survey". IEEE Trans. . PAMI,vol 24(01), pp 34–58, 2002.
[6] E. Hjelmas and B.K. Low."Face detection: A survey". Computer Vision and Image Understanding,vol 83 ,pp 236–274, Edinburgh ,Scotland,UK, April 17,2001. //article
[7] Cheng-Chin, Wen-Kai Tai, Mau-Tsuen Yang, Yi-Timg Huang, and chi-Janng Huang. A novel method for detecting lips, eyes and faces in real time. Real-Time Imaging, vole 9(4) : pp277-287, 2003.//article
[8] https://www.math.univ-toulouse.fr/~besse/Wikistat/pdf/st-m-Intro-ApprentStat.pdf
[9] Ming-Hsuan Yang, David J. Kriegman et Narendra Ahuja. Detecting faces images : A survey. Dans IEEE Transaction on Pattern Analysis and Machine Intelligence, vol.24(1),pp34-58 ,Taiwane, 2002,
[10] Wenlong Zheng and Suchendra M. Bhandarkar. Face detection and haking using a boosted adaptative particle filter. Journal of visual communication and Imog Representation,vole 20(1) :pp 9-27,US,2009.
[11] H. A. Rowley, S. Baluja, and T. Kanade. Neural Network-based face detection. IEEE Transachen on Pattern Analysis and Machine Intelligence,vole 20(1) :pp 23-38, China,1998.
[12] H. Schneidermand and T. Kanade. Probabilistic modeling of local appearance and spacial relation ships for object recognition. Computer Vision and Pattern Recognition, IEEE Computer Society Conférence ,:pp 45,Santa Barbara, July 1998.
[13] Paul Viola and Michael Jones. Robust real-time object detection. In Second international work shop on statistical and computation atheories of vision, Vancouver, Canada, July 13 2001.
[14] Viola. P, Jones. M, "Rapid object detection using a boosted cascade of simple features", Proceedings of the IEEE Computer Society Conference, vol 1, pp 511-518,Hawaii,December 2001
[15] Viola. P, Jones. M, "Robust Real-time Object Detection", Second international workshop on statistical and computational theories of vision - modèleing, learning, computing, and sampling. Vancouver, Canada, July 13, 2001.
[16] Viola. P, Jones. M, "Robust real-time face detection", International Journal of Computer Vision vole 57 (2), pp 747-747, Dublin,Ireland,July 2001.
[17] M. Jones et P. Viola, Fast Multi-View Face Detection, IEEE CVPR, 2003.
[18] A. L. C. Barzak, To Ward and Efficient Implementation of a Rotation Invariant Detection using Haar-like Features. Proceeding of the 2009 IEEE/RSJ international conference on Intelligent robotsand systems.pp31-36. Nouvelle-Zélande, Dunedin. 2005,

[19] M. Kolsch et M. Turk, Analysis of Rotational Robustness of Hand Detection with a Viola-Jones Detector, ICPR, Vol.3, California,Santa Barbara,2004.
[20] T. COIANIZ, L. TORRESANI, B. CAPRILE, «2D Deformable Models for Visual Speech Analysis ». In NATO Advanced Study Institute : Speech reading by Man and Machine,
[21] The Database of Faces, Cambridge University Department, http://www.uk.research.att.com/facedatabase.html
[22] Y. TIAN, T. KANADE, J. COHN, « Robust Lip Tracking by Combining Shape, Color and Motion ». Proc ACCV'00, 2000
[23] Y. TIAN, T. KANADE, and J. COHN, « Dual state Parametric Eye Tracking ». Proceedings of the 4th IEEE International Conference on Automatic Face and Gesture Recognition , pp. 110- 115,Grenoble, France, March 2000
[24] G.C. Feng and P.C. Yuen. Multi-cues eye detection on gray intensity image. Pattern Recognition, 34(5) :1033_1046, 2001.
[25] B.S. Venkatesh, S. Palanivel, and B. Yegnanarayana. Face detection and recognition in an image sequence using eigenedginess. In Proc. of Indian Conference on Vision Graphics and Image Processing, pages 97_101, 2002.
[26] J. Wu and Z.H. Zhou. E_cient face candidates selector for face detection. Pattern Recognition, 36(5) :1175_1186, 2003.
[27] H. Byun and B. Ko. Robust face detection and tracking for real-life applications. Int. Journal of Pattern Recognition and Arti_cial Intelligence, 17(6) :1035_1055, 2003.
[28] C.C. Chiang, W.K. Tai, M.T. Yang, Y.T. Huang, and C.J. Huang. A novel method for detecting lips, eyes and faces in real time. Real-Time Imaging, 9(4) :277_287, 2003.
[29] T.E. Campos, R.M. Cesar, and R.S. Feris. Detection and tracking of facial features in video sequences. In Proc. of the Mexican Int. Conference on Arti_cial Intelligence, volume 1793, pages 127_135, 2000.
[30] M.A. Bhuiyan, V. Ampornaramveth, S.Y. Muto, and H. Ueno. Face detection and facial feature localization for human-machine interface. NII Journal, 5 :25_39, 2003.
[31] R. Lanzarotti, N.A. Borghese, and P. Campadelli. Automatic features detection for overlapping face images on their 3D range models. In Proc. of Int. Conference on Image Analysis and Processing, pages 316_321, 2001.
[32] A. Senior, R.L. Hsu, M.A. Mottaleb, and A.K. Jain. Face detection in color images. IEEE Trans. on Pattern Analysis and Machine Intelligence, 24(5) :696_706, 2002.
[33] V. Vezhnevets, S. Soldatov, and A. Degtiareva. Automatic extraction of frontal facial features. In Proc. of the Asian Conference on Computer Vision (ACCV'04), volume 2, pages 1020_1025, 2004.
[34] A.M. Alattar and S.A. Rajala. Facial features localization in front view head and shoulders images. In Proc. of Int. Conference on Acoustics, Speech and Signal Processing, volume 6, pages 3557_3560, 1999.
[35] G.G. Mateos and C.V. Chicote. A uni_ed approach to face detection segmentation and location using hit maps. In Spanish Symposium on Pattern Recognition and Image Analysis, 2001.
[36] K.H. Lin, K.M. Lam, and W.C. Siu. Locating the eye in human face images using fractal dimensions. IEE Proc. Vision, Image and Signal Processing, 148(6) :413_421, 2001.

[37] Y. Tian, T. Kanade, and J.F. Cohn. Multi-state based facial features tracking and detection. Technical report, Technical report CMU-RI-TR-99-18, Robotics Institute, Carnegie Mellon University, August 1999.
[38] Y. Nawaz and S. Sircar. Real time eye tracking and blink detection using low resolution webcam. In Newfoundland Electrical and Computer Engineering Conference (NECEC'03), 2003.
[39] T. Kawaguchi and M. Rizon. Iris detection using intensity and edge information. Pattern Recognition, 36(2) :549_562, 2003.
[40] M. Lievin, P. Delmas, P.Y. Coulon, F. Luthon, and V. Fristot. Automatic lip tracking :Bayesian segmentation and active contours in a cooperative scheme. In Proc. of Int. Conference on Multimedia Computing and Systems (ICMCS'99), volume 1, pages 691_ 696, 1999.
[41] M. Lievin and F. Luthon. Nonlinear color space and spatiotemporal MRF for hierarchical segmentation of face features in video. IEEE Trans. on Image Processing, 13(1) :63_71, 2004.
[42] C.H. Lin and J.L. Wu. Automatic facial feature extraction by genetic algorithms. IEEE Trans. on Image Processing, 8(6) :834_845, 1999.
[43] M.J. Reinders, R.W. Koch, and J.J. Gerbrands. Locating facial features in image sequences using neural networks. In Proc. of the Second Int. Conference on Automatic Face and Gesture Recognition, pages 230_235, 1996.
[44] T. K. Leung, M.C. Burl, and P. Perona. Finding faces in cluttered scenes using random labeled graph matching. In Proc. of the Fifth Int. Conference on Computer Vision, pages 637_644, 1995.
[45] K.C. Yow and R. Cipolla. Feature-based human face detection. Image and Vision Computing, 15(9) :713_735, 1997.
[46] B. Moghaddam and A. Pentland. Probabilistic visual learning for object recognition. IEEE Trans. on Pattern Analysis and Machine Intelligence, 19(7) :696_710, 1997.
[47] R.S. Feris, J. Gemmell, K. Toyama, and V. Krüger. Hierarchical wavelet networks for facial feature localization. In Proc. of the Fifth IEEE Int. Conference on Automatic Face and Gesture Recognition, 2002.
[48] P. Sozou, T. Cootes, and C. Taylor. A non-linear generalisation of point distribution models using polynomial regression. In British Machine Vision Conference, pages 397_ 406, 1994.
[49] T. Cootes and C. Taylor. Active Shape Model search using local grey-level models : A quantitative evaluation. In British Machine Vision Conference, pages 639_648, 1993.
[50] T. Cootes, G. Edwards, and C. Taylor. Active Appearance Models. IEEE Trans. On Pattern Analysis and Machine Intelligence, 23(6) :681_685, 2001.
[51] D. Cristinacce and T. Cootes. A comparison of shape constrained facial feature detectors. In Proc. of the 6th Int. Conference on Automatic Face and Gesture Recognition, pages 375_380, Seoul, Korea, 2004.

[52] https://fr.wikipedia.org/wiki/R%C3%A9seau_de_neurones_artificiels
[53] https://fr.wikipedia.org/wiki/Fonction_d%27activation
[54] Pierre Buyssens, Fusion de différents modes de capture pour la reconnaissance du visage appliquée aux e_ transactions DOCTORAT de l'UNIVERSIT´E de CAEN Janury-4-2011
[55] http://www.statsoft.fr/concepts-statistiques/reseaux-de-neurones automatises/reseaux-de-neurones-automatises.htm
[56] D. H. Hubel and T. Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. Physiol ,vole160(1) ,Debrecen,Hungary,pp 106–154, 1962.
[57] https://msdn.microsoft.com/big_data_france/2014/06/17/evaluer-un-modle-en-apprentissageautomatique/ recognition. Proceedings of IEEE, vole 86(11) :pp 2278–2324, November 1998
[58] Yann LeCun, Leon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. Proceedings of the IEEE, vole 86(11) :pp 2278–2324, 1998.
[59] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems, pp 1097–1105, 2012.
[60] https://fr.wikipedia.org/wiki/R%C3%A9tropropagation_du_gradient
[61] https://fr.wikipedia.org/wiki/R%C3%A9seau_neuronal_convolutif
[62] https://fr.wikipedia.org/wiki/R%C3%A9seau_neuronal_convolutif
[63] A. Krizhevsky, I. Sutskever et G.E. Hinton. Imagenet classi- fication with deep convolutional neural networks. NIPS, 2012.
[64] https://www.theses.fr/2017CLFAC018.pdf
[65] K. He, X. Zhang, S. Ren et J. Sun. Deep Residual Learning for Image Recognition. CVPR, 2016
[66] K. Simonyan et A. Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. ICLR, 2014.
[67] https://www.pyimagesearch.com/20/03/2017/imagenet-vggnet-resnet-inception-xception-keras/
[68] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke et A. Rabinovich. Going Deeper with Convolutions. CVPR, 2015.
Chap 4
[69] N. Dalal et B. Triggs. "Histograms of oriented gradients for human detection ''.In CVPR, France, 2005.
[70] N. Dalal. "Finding People in Images and Videos''. Thèse de doctorat, L'institut National Polytechnique de Grenoble,university of Monastir,2016.
[71] Erjin Zhou et al. (2013): "Extensive Facial Landmark Localisation with coarse to-fine Convolutional Network Cascade".