

République Algérienne Démocratique et Populaire

Ministère de l'enseignement supérieur et de la recherche scientifique

Université de Guelma

Faculté des Mathématiques, d'Informatique et des Sciences de la matière

Mémoire de fin d'étude Master



Département d'Informatique

Spécialité : Ingénierie des média

Thème :

**Amélioration des performances d'un système de
détection de visages sur flux vidéo**

**Présenté par : BAHNOUS ATHMAN
BRAGHTA ISMAIL**

Sous la direction de :

Dr BENCHERIET CHAMESSENNEHAR

Juin 2011

Table des matières

| | |
|--|----|
| Introduction générale | 1 |
| Chapitre 01 : Biométrie et détection de visage | |
| I. Introduction | 3 |
| II. Historique | 3 |
| III. Les systèmes biométriques et le visage | 3 |
| III.1 La biométrie..... | 3 |
| III.2 Quelques systèmes biométriques | 4 |
| III.3 Le visage humain | 4 |
| IV. Détection de visage | 4 |
| IV.1 Définition | 4 |
| IV.2 Problèmes de détection de visage..... | 5 |
| IV.3 Prétraitement nécessaire dans le système de détection de visage | 8 |
| IV.4 Méthodes de détection de visage | 8 |
| V. Comparaison des méthodes de détection de visages | 15 |
| VI. Suivi du visage dans une vidéo numérique | 16 |
| IV.1 La vidéo | 16 |
| IV.2 Formats vidéo | 16 |
| IV.3 La détection des points d'intérêt | 17 |
| IV.4 Méthodes de suivi des points d'intérêt | 18 |
| VII. Conclusion | 19 |
| Chapitre 02 : Les espaces de couleur | |
| I. Introduction | 20 |
| II. La perception de la couleur et la théorie trichromatique | 20 |
| II.1 Espaces de la couleur | 21 |
| III. Détection de peau dans une image couleur..... | 25 |
| III.1 Choix de l'espace couleur | 25 |

| | | |
|-------|-------------------------------|----|
| III.2 | Modélisation de la peau | 26 |
| IV. | Conclusion | 28 |

Chapitre 03 : Les réseaux de neurones

| | | |
|------|--|----|
| I. | Introduction | 29 |
| II. | Historique | 29 |
| III. | Avantage par rapport aux méthodes classiques | 30 |
| IV. | Réseaux de neurone | 30 |
| IV.1 | Le neurone biologique | 30 |
| IV.2 | Le neurone Formel | 33 |
| V. | Le perceptron multicouche | 41 |
| V.1 | Définition | 41 |
| V.2 | Algorithme de rétro propagation | 42 |
| V.3 | Propriétés des RNA | 42 |
| V.4 | Domaine d'applications | 42 |
| VI. | Conclusion | 43 |

Chapitre 04 : Conception & implémentation

| | | |
|-------|--|----|
| I. | Introduction | 44 |
| II. | Conception | 44 |
| II.1 | Stockage dans un buffer | 45 |
| II.2 | Chargement d'une image | 45 |
| II.3 | Segmentation de peau | 45 |
| II.4 | Segment peau..... | 49 |
| II.5 | Application de réseau de neurone | 53 |
| III. | Implémentation | 55 |
| III.1 | Environnement de développement | 55 |
| IV. | Base de données image | 58 |
| V. | Présentation de l'application..... | 60 |

| | | |
|------|----------------------------------|----|
| VI. | Test et résultat | 65 |
| VI.1 | Test sur image fixe | 65 |
| VI.2 | Test sur vidéo archivée | 69 |
| VI.3 | Test vidéo en direct | 71 |
| VII. | Conclusion | 71 |
| | Conclusion générale | 72 |

LISTE DES FIGURES

| | |
|--|----|
| Figure 1.1 : Problème de position du visage | 5 |
| Figure 1.2 : Problème de la couleur du visage | 5 |
| Figure 1.3 : Problème de la taille de visage | 6 |
| Figure 1.4 : Présence ou pas de quelques composants du visage | 6 |
| Figure 1.5 : Problème d'Occultation | 6 |
| Figure 1.6 : Condition d'illumination différente | 7 |
| Figure 1.7 : Problème de rotation | 7 |
| Figure 1.8 : Problème de complexité de l'image | 8 |
| Figure 1.9 : De gauche à droite : l'image brute, l'image rendue plus lisible, et en fin avec l'amélioration de netteté du fond | 8 |
| Figure 1.10 : Structure d'un neurone artificiel | 9 |
| Figure 1.11 : Détection basée sur les caractéristiques bouche et l'œil | 12 |
| Figure 1.12 : Le visage typique de Yang et Huang | 13 |
| Figure 1.13 : Détection de la couleur de la peau dans divers images | 14 |
| Figure 1.14 : a) Pattern de visage avec égalisation d'histogramme. b) Pattern de visage après élimination du fond | 15 |
| Figure 1.15 : Exemples de codes vidéo avec domaines d'application | 17 |
| Figure 1.16 : Exemple de détection des points d'intérêt. De gauche à droite : première image de la séquence ; les zones d'intérêt, yeux et nez, détectées de manière automatique | 18 |
| Figure 1.17 : Principe de la méthode KLT. On recherche la transformation W qui minimise la somme des erreurs quadratiques | 18 |
| Figure 2.1 : (a) Histogramme des pixels de peau (b) modèle gaussienne de peau dans l'espace RGB | 27 |
| Figure 2.2 : (a) Histogramme des pixels de peau (b) modèle gaussienne de peau dans l'espace YCrCb | 28 |
| Figure 2.3 : (a) Histogramme des pixels de peau (b) modèle gaussienne de peau dans l'espace HSI | 28 |
| Figure 3.1 : Neurone biologique | 31 |
| Figure 3.2 : Fonctionnement de neurone biologique | 32 |
| Figure 3.3 : Réseau de neurones artificiels | 33 |
| Figure 3.4 : Somation du RNA | 34 |
| Figure 3.5 : Réseau multicouches | 36 |

| | |
|--|----|
| Figure 3.6 : Réseau à connexions locales | 37 |
| Figure 3.7 : Réseau à connexions récurrentes | 37 |
| Figure 3.8 : Réseau à connexion complète | 38 |
| Figure 3.9 : Réseau de type perceptron multicouche | 41 |
| Figure 4.1 : Etapes de la conception | 44 |
| Figure 4.2 : Processus de détection de peau | 46 |
| Figure 4.3 : La base d'échantillons peau | 47 |
| Figure 4.4 : Model gaussien de peau dans l'espace YCbCr | 48 |
| Figure 4.5 : Application du modèle gaussien simple | 48 |
| Figure 4.6 : Application du modèle mélange gaussien | 49 |
| Figure 4.7 : Prétraitement des segments de peau | 49 |
| Figure 4.8 : Application de la dilatation | 50 |
| Figure 4.9 : Masques de voisinage de e pour : (a) le parcours avant ; (b) le parcours arrière | 51 |
| Figure 4.10 : Les résultats de l'étiquetage | 51 |
| Figure 4.11 : Application des contraintes de surface et de ratio | 52 |
| Figure 4.12 : Extraction des visages probables | 53 |
| Figure 4.13 : Le processus de la décision visage/non visage par PMC | 53 |
| Figure 4.14 : échantillons de la base d'apprentissage | 54 |
| Figure 4.15 : Application du réseau de neurone | 55 |
| Figure 4.16 : L'interface de C++ builder | 57 |
| Figure 4.17 : Le composant SDL | 58 |
| Figure 4.18 : Les composants videolab | 58 |
| Figure 4.19 : Quelques échantillons de la base « BAO Faces » | 59 |
| Figure 4.20 : Quelques échantillons de la base « Caltech Faces » | 59 |
| Figure 4.21 : Quelques échantillons de la base « Valid Faces » | 60 |
| Figure 4.22 : Fenêtre de lancement de « DFACE » | 60 |
| Figure 4.23 : Fenêtre principale de « DFACE » | 61 |
| Figure 4.24 : Fenêtre due au traitement sur image fixe | 62 |
| Figure 4.25 : Fenêtre d'apprentissage de réseau de neurones | 62 |
| Figure 4.26 : Fenêtre de traitement sur les vidéos | 63 |
| Figure 4.27 : Fenêtre de traitement en direct | 64 |
| Figure 4.28 : Fenêtre à-propos | 64 |
| Figure 4.29 : Test 01[acquise par webcam] | 65 |
| Figure 4.30 : Test 02[base caltetch] | 66 |
| Figure 4.31 : Test 03[base valid] | 67 |
| Figure 4.32 : Test 04[base BAO] | 67 |
| Figure 4.33 : Test 05[base caltech] | 68 |

| | |
|---|-----------|
| Figure 4.34 : <i>Test 06[base valid]</i> | 69 |
| Figure 4.35 : <i>Test 07</i> | 70 |
| Figure 4.36 : <i>Test 08</i> | 70 |

Liste des tableaux

| | |
|---|-----------|
| Tableau 1.1 : Comparaison entre les différentes méthodes de détection de visage..... | 15 |
| Tableau 4.1 : Le taux d'apprentissage pour 500 itérations..... | 54 |
| Tableau 4.2 : Le taux d'apprentissage pour 5000 itérations..... | 54 |
| Tableau 4.3 : Résultats de Test 01..... | 65 |
| Tableau 4.4 : Résultats de Test 02..... | 66 |
| Tableau 4.5 : Résultats de Test 03..... | 67 |
| Tableau 4.6 : Résultats de Test 04..... | 67 |
| Tableau 4.7 : Résultats de Test 05..... | 68 |
| Tableau 4.8 : Résultats de Test 06..... | 69 |
| Tableau 4.9 : Résultats de Test 07..... | 70 |
| Tableau 4.10 : Résultats de Test 08..... | 70 |
| Tableau 4.11 : Résultats de Test 08..... | 71 |

Remerciement

Nous remercions ALLAH de nous avoir éclairé le chemin du s'avoir, et de nous avoir armé de foi, de patience et de force afin d'élaborer ce travail.

Nous remercions :

Dr Bencheriet notre encadreur de l'aide qui nous a apporté.

*Ainsi que Tous les enseignants qui furent nos guidés
durant notre parcours universitaire.*

Résumé

La détection de visage est un problème intéressant développé depuis plus d'une dizaine d'années. Le problème de détection de visages est un problème classique de reconnaissance de formes qui peut s'énoncer de la façon suivante : considérant une image d'entrée quelconque, qu'elle soit scannée ou extraite d'un flux vidéo, il s'agit de déterminer la présence ou non d'un ou plusieurs visages humains dans cette image.

Le travail ici présent est une amélioration des performances d'un système de détection de visages sur un flux vidéo réalisé dans le cadre d'un PFE de master II réalisé en 2010.

Ce dernier ayant été conçu sous matlab présentait l'inconvénient du temps d'exécution élevé surtout que l'objectif principal et de travaillé en temps réel sur un flux vidéo .Pour remédier à ce problème nous avons implémentés notre système « DFACE » sous C++ Builder (renommé par ces performances en terme temps de traitement).

Le système « DFACE » est basé sur une hybridation de deux méthodes : la mixture gaussienne et réseau neuronal, La première consiste à détecter les zones contenant la couleur chair humaine par mixture de modèles gaussien l'objectif principal de cette étape est la limitation de l'espace de recherche de l'objet visage. La seconde consiste à décider de l'appartenance de la zone peau à la classe des visages ou des non visages par le Perceptron Multicouches (PMC).

Mots-clés :

Détection de visage, couleur chair, mixture de modèles gaussiens, modèle gaussien, réseau de neurone artificiel, image vidéo.

Introduction générale

Ces dernières années, divers d'applications ont été conçu dans le cadre de la détection de visages dans une image ou dans une séquence vidéo, le problème de détection de visages est un problème classique de reconnaissance de formes qui peut s'énoncer de la façon suivante : considérant une image d'entrée quelconque, qu'elle soit scannée ou extraite d'un flux vidéo, il s'agit de déterminer la présence ou non d'un ou plusieurs visages humains dans cette image.

Plusieurs méthodes ont été proposée pour résoudre ce problème, parmi les méthodes existantes, on cite : l'approche la plus réponde est sans doute, celle basée sur les réseaux de neurones et l'approche basée sur la détection des points d'intérêts, mais aucune de ces méthodes ne s'avère être sans erreurs.

L'objectif de ce travail consiste à améliorer les performances d'un système de détection de visages sur un flux vidéo. Le système Original ayant été conçu sous Matlab souffrait de temps d'exécution élevé surtout que l'objectif principal était de travailler en temps réel.

Comme solution à ce problème. Nous avons implémenté Ce logiciel Sous C++ Builder.

Le nouveau Système (que nous avons nommé DFACE) est constitué de deux principaux Modules : le premier, basé sur un Mélange de Modèles Gaussiens, a comme tâche principale la détection de la peau sur le flux d'images, et le second basé sur les réseaux de neurones, décide de l'appartenance des segments peau à la classe visage ou non visage.

~~Notre mémoire est composé de quatre chapitres :~~

Amélioration des performances d'un système de détections de visages sur flux vidéo

Chapitre 01 :

Dans ce chapitre nous abordons la notion de biométrie et les systèmes biométriques, suivie d'une étude détaillée sur la détection de visages, les problèmes rencontrés ainsi que les méthodes de détection les plus répandues et une comparaison entre ces derniers.

Chapitre02 :

En plus d'une description détaillée de la perception de la couleur, ce chapitre a fait l'objet d'une illustration sur la modélisation de la peau et sa distribution dans l'espace de chrominance.

Chapitre 03 :

Nous avons réservés ce chapitre à un état de l'art sur les réseaux de neurones : naissance et analogie avec les biologiques, topologies et apprentissage.

Ici on explique les principaux avantages du réseau de neurone ainsi leur fonctionnement.

Le Chapitre 04 :

Illustre :

- Dans sa première partie, la conception de notre logiciel et les méthodes utilisées
- Dans la deuxième partie, l'implémentation et la réalisation du projet, les tests réalisés, et les résultats obtenus avec leurs interprétations.

- I. Introduction
- II. Historique
- III. Les systèmes biométriques et le visage
- IV. Détection de visage
- V. Comparaison des méthodes
- VI. Suivi du visage dans une vidéo numérique
- VII. Conclusion

I. Introduction

La détection de visage est un domaine de la vision par ordinateur consistant à détecter un visage humain dans une image numérique. C'est un cas spécifique de détection d'objet, où l'on cherche à détecter la présence et la localisation précise d'un ou plusieurs visages dans une image. C'est l'un des domaines de la vision par ordinateur parmi les plus étudiés, qui trouve son application dans différentes disciplines tel que : vidéosurveillance, biométrie, robotique, commande d'interface homme-machine, photographie, indexation d'images et de vidéos, recherche d'images par le contenu, ...etc.

II. Historique

L'utilisation du visage comme outil biométrique a commencé dans les années 70. Le domaine de l'analyse et de la compréhension du visage a attiré plusieurs chercheurs venants de plusieurs disciplines telles que : le traitement du signal et le traitement d'image, les réseaux de neurones, la reconnaissance des formes, la psychologie...etc.

L'analyse et la compréhension du visage a connu ensuite quelques déclis dans les années 80 à cause du coût de conception élevé à cette époque. C'est aux années 90 que cette dernière a pris une grande ampleur pour plusieurs raisons dont on peut citer :

- Les opportunités commerciales qui ont sensiblement augmenté.

- Le hardware est devenu plus rapide et moins coûteux, ce qui a permis d'avoir des systèmes en temps réel avec peu d'investissements.
- Les applications liées à la surveillance ont accru après la diffusion surprenante et rapide de l'informatique.[1]

III. Les systèmes biométriques et le visage

III.1 La biométrie

La biométrie est définie comme la science permettant l'identification d'individus à partir de leurs caractéristiques physiologiques ou comportementales. Ces caractéristiques n'ont pas le même degré de fiabilité, mais ils doivent satisfaire les conditions suivantes :

- **Universelles** : caractéristiques présentes chez tous les individus.
- **Unicité** : caractéristiques permettant de différencier un individu par rapport à tout autre.
- **Permanentes** : présentes tout au long de la vie.
- **Collectables**: possibilité d'enregistrer les caractéristiques d'un individu avec son accord.
- **Mesurables** : autorisant une comparaison future.

Donc, l'avantage principal de la biométrie est que chaque personne a ses propres caractéristiques qui ne peuvent pas être subtilisées ou détournées.[2]

III.2 Quelques systèmes biométriques

Grâce aux nombreuses caractéristiques physiques de l'être humain, il existe aujourd'hui plusieurs méthodes qui servent à l'identification ou à l'authentification d'une personne dans un système. Cependant nous pouvons distinguer deux grandes catégories de méthodes d'authentification biométrique :

- Méthodes basées sur les caractéristiques morphologiques. - Méthodes basées sur les caractéristiques comportementales. [2]

III.3. Le visage humain

Le visage humain est le premier point d'intérêt dans le cadre d'une communication entre personnes, car il est le support du regard, de la représentation des émotions, et également le

Amélioration des performances d'un système de détections de visages sur flux vidéo

centre d'émission de la parole. Le visage présente le grand avantage d'avoir une structure fixe semblable entre des personnes différentes, relativement peu déformable. Cette structure peut être utilisée pour détecter un visage ou le suivre. [3]

III. Détection de visage

IV.1 Définition

La détection du visage est la première phase d'un système biométrique basé sur la reconnaissance du visage. Le but est de vérifier la présence d'un ou plusieurs visages sur une image et retourner leurs positions, puis généraliser le traitement sur une séquence vidéo.[1]

IV.2 Problèmes de détection de visage

La détection du visage est une tâche très facile pour l'être humain: L'image est capturée par l'œil qui la transfère vers le cerveau pour analyse, afin de vérifier la présence des visages et de les localiser. Cette même tâche s'avère très complexe pour un ordinateur et cela pour de nombreuses raisons.[1]

IV.2.1 La position

Il s'agit des différentes positions du visage dans une image, dans ce cas l'ordinateur doit pouvoir détecter le visage quelque soit sa position sur l'image.



Figure 1.1 : Problème de position du visage.[1]

IV.2.2 Couleur du visage

Les êtres humains ont des couleurs de peau différentes, d'où la différence de la valeur du pixel représentant la peau de chaque personne.



Figure 1.2 : Problème de la couleur du visage.[1]

IV.2.3 Taille

La taille des visages est différente d'une image à une autre ou d'une personne à une autre d'où la difficulté de l'implémentation d'un algorithme qui détecte les visages sans avoir de conséquences de ce facteur. Il y a aussi la taille des composants du visage tel que le nez, les yeux, la bouche ou autre chose variant d'une personne à une autre ; ce qui implique un plus grand nombre de paramètres lors de la réalisation de la détection.



Figure 1.3 : Problème de la taille de visage.[1]

IV.2.4 Présence ou pas de quelques composants du visage

Il s'agit des moustaches, de la barbe, des lunettes...etc. ces composantes doivent être prises en considération lors de l'implémentation de l'algorithme.



Figure 1.4 : Présence ou pas de quelques composants du visage.

IV.2.5 Occultation

Un visage qui peut apparaître à moitié dans une image ou parfois masqué partiellement par un objet nous oblige à définir des conditions d'acceptation du visage par le système. Par exemple, on peut supposer que le visage doit apparaître entièrement pour qu'il soit admis.



Figure 1.5 : Problème d'Occultation.[1]

IV.2.6 Les conditions d'éclairage et d'illumination

Dans toute action de détection, la lumière est un facteur important et c'est le problème le plus délicat à résoudre. Il s'est avéré qu'on ne peut réaliser un système fiable sans prendre ce facteur en considération; d'où la nécessité de faire des prétraitements de l'image comme la normalisation et l'égalisation d'histogramme afin de minimiser les effets d'éclairage et d'illumination.

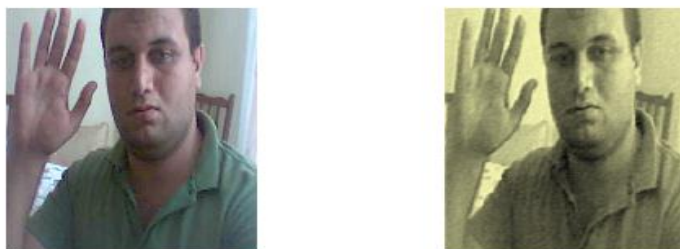


Figure 1.6 : Condition d'illumination différente.[1]

IV.2.7 Rotation

Les visages ne sont pas toujours face à la caméra. Certains visages présentent une inclinaison avec un certain degré, ce qui influe sur le système de détection d'où la nécessité d'établir des conditions pour définir un visage candidat pour l'acceptation.



Figure 1.7 : Problème de rotation.[1]

IV.2 .8 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.



Figure 1.8 : Problème de complexité de l'image.[1]

IV.3 Prétraitement nécessaire dans le système de détection de visage

Le système accepte une image d'entrée soit une image couleur, soit une image de niveaux de gris. Avant de les traiter, quelques opérations de prétraitement telles que le réhaussement du contraste, l'égalisation de l'histogramme et l'élimination du bruit sont utilisées pour améliorer leur qualités. [4]

Actuellement les services de sécurité opérationnels sont demandeurs de technologies capables d'améliorer les images, pour que les fonctions d'analyse aient de meilleurs résultats (Figure 1.9) [5] :



Figure 1.9 : De gauche à droite : l'image brute, l'image rendue plus lisible, et en fin avec l'amélioration de netteté du fond[5]

IV.4 Méthodes de détection de visage

De nombreuses méthodes entrent dans l'approche basée sur image. Nous pouvons

subdivisée cette approche en trois principal méthodes : la méthodes d'apprentissage statistique ,la méthodes statistiques réduisant l'espace des descripteurs et la méthodes basées sur la géométrie .

IV.4.1 Méthodes d'apprentissage statistique

C'est des méthodes dont le principe est de construire un classificateur statistique capable de prendre une décision sur une image de même taille que l'image de formation pour l'associer à la classe des images visages ou à la classe des images non visages. Dans cette catégorie de méthodes on trouve : les Réseau de neurones, le Modèles de Markov Cachés et les Machines à Vecteur de Support (SVM).

IV.4.1.1 Réseau de neurones :

Les réseaux de neurones artificiels sont des réseaux fortement connectés de processeurs élémentaires fonctionnant en parallèle. Chaque processeur élémentaire calcule une sortie unique sur la base des informations qu'il reçoit. Toute structure hiérarchique de réseaux est évidemment un réseau.

Dans un réseau de neurones, deux modes de fonctionnement peuvent être distingués. Dans le premier, les paramètres du réseau sont ajustés grâce à la présentation d'exemples pour lesquels on connaît la réponse désirée : ce mode de fonctionnement est appelé « phase d'apprentissage ». Dans le deuxième mode, il s'agit d'exploiter le réseau en lui présentant des données inconnues: ce mode d'utilisation est souvent appelé « phase de reconnaissance », ou « phase de généralisation ».

Amélioration des performances d'un système de détections de visages sur flux vidéo

Les réseaux de neurones les plus répandus et les plus simples à la fois restent les Perceptrons multicouches (PMC) qui consistent à une succession de 3 couches, interconnectées totalement ou partiellement. [6]

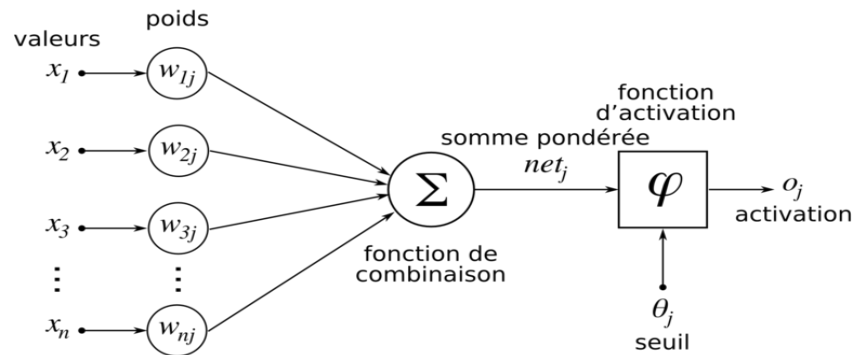


Figure 1.10 : Structure d'un neurone artificiel.[6]

IV.4.1.2 Machines à Vecteur de Support (SVM)

SVM (Support Vector Machines) est une technique d'apprentissage statistique, proposée par V. Vapnik en 1995. Elle permet d'aborder des problèmes très divers comme la classification, la régression, la fusion,... etc.

Depuis son introduction dans le domaine de la Reconnaissance de Formes (RDF), plusieurs travaux ont pu montrer l'efficacité de cette technique principalement en traitement d'images .

L'idée essentielle consiste à projeter les données de l'espace d'entrée (appartenant à des classes différentes) non linéairement séparables, dans un espace de plus grande dimension appelé espace de caractéristiques, de façon à ce que les données deviennent linéairement séparables .Dans cet espace, la technique de construction de l'hyperplan optimal est utilisée pour calculer la fonction de classement séparant les classes.[8]

IV.4.1.2 Modèles de Markov Cachés (HMM)

Les HMM sont un ensemble de modèles statistiques utilisés pour caractériser les propriétés statistiques d'une image. L'image est divisée en N régions significatives où chaque région est assignée à un état S_i qui est caractérisé par une fonction de probabilités, estimée

sur les bases des images de formation. Dans un visage les cheveux, le front, les yeux, le nez et la bouche peuvent former cinq états distincts.

Pour pouvoir détecter et localiser un visage, l'algorithme HMM extrait les mêmes régions de l'image et vérifie l'équivalence de l'ordre d'apparition des objets caractéristiques avec l'ordre défini dans le modèle HMM. [9]

IV.4.2 Méthodes statistiques réduisant l'espace des descripteurs

C'est des techniques dont le rôle est de réduire la dimension de l'espace de l'image et d'effectuer une décorrélation des images de formation. Ces méthodes sont appelées méthodes d'extraction de paramètres. Le système de détection est conçu en implémentant une technique d'apprentissage statistique ou en combinant une méthode d'apprentissage statistique avec une méthode d'extraction de paramètres. Dans cette catégorie de méthodes on trouve : la méthode d'analyse en Composantes Principales (PCA) et la méthode d'analyse Discriminante de Fisher (LDA). L'utilisation des méthodes réduisant l'espace des descripteurs sert à améliorer le temps et la précision de la détection. [1]

IV.4.2.1 L'analyse en Composantes Principales (PCA)

L'analyse en composantes principales a été présentée par MP. Pentland et MA. Turc en 1991. La PCA ne nécessite aucune connaissance a priori sur l'image.

Le principe est qu'on peut construire un sous-espace vectoriel en ne retenant que les meilleurs vecteurs propres, tout en conservant beaucoup d'information utile non redondante (parfaitement corrélée).

C'est pourquoi la PCA est très efficace pour réduire la dimension des données. On peut toujours utiliser la PCA pour classifier les visages et les non visages mais cela reste peu performant comparé à une méthode d'apprentissage statistique. [1]

IV.4.2.2 L'analyse Discriminante de Fisher (LDA)

En 1997 Belhumeur de la Yale Université (USA) a mis au point l'algorithme LDA. Contrairement à l'algorithme PCA, l'algorithme LDA effectue une véritable séparation des classes.

La LDA analyse les vecteurs propres de la matrice de dispersion des données et cela selon un

critère mathématique dont le principe est de minimiser les variations entre les images d'une même classe (variations intra-classe) tout en maximisant les variations entre les images positives et négatives (variations interclasses). La fusion entre la PCA et la LDA a donnée des résultats meilleurs que la PCA ou la LDA seule. [1]

IV.4.3 Méthodes basées sur la géométrie (feature-based)

Contrairement à la précédente approche, les méthodes appartenant à cette approche nécessitent des connaissances préalables sur les objets caractéristiques pour pouvoir détecter et localiser les visages. Les connaissances préalables peuvent être un ensemble de caractéristiques quantifiables comme la distance entre la bouche et l'œil droit (Figure 11). Dans cette catégorie de méthodes on trouve : les méthodes basées sur les caractéristiques faciales d'un visage type, les méthodes basées sur les caractéristiques invariables et les méthodes basées sur la mise en correspondance. [1]

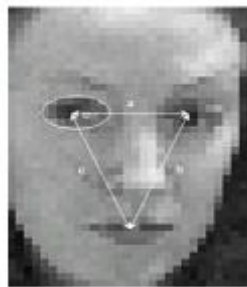


Figure 1.11 : Détection basée sur les caractéristiques bouche et l'œil. [1]

IV.4.3.1 Méthodes basées sur les caractéristiques faciales d'un visage type

Les méthodes utilisées dans cette approche se basent sur les caractéristiques faciales. Pour cela un visage type est défini. Les caractéristiques faciales peuvent être par exemple les yeux, la bouche et le nez. Ce qui nous intéresse c'est le rapport entre ces caractéristiques qui peuvent être la position de ces organes ou la distance entre eux par exemple. On calcul le rapport des caractéristiques faciales du visage type. Puis on extrait les caractéristiques faciales des images en entrée et on calcul leur rapports, ces rapports sont comparés un par un avec le rapport correspondant du visage type à un seuil près.

✓ L'algorithme de détection est le suivant :

- Si le premier rapport est égal à un seuil près au premier rapport du visage type, cela nous amène à comparer le rapport suivant sinon l'image analysée n'est pas une image de

visage.

- Si à la fin tous les rapports sont égaux, l'image analysée est une image de visage.

La difficulté dans cette approche est l'obtention de règles bien définies :

- Si les règles sont détaillées, elles peuvent ne pas détecter quelques visages qui ne vérifient pas toutes les règles imposées.
- Si les règles sont très générales, on peut détecter des images qui ne sont pas des images représentant des visages.

Le fait d'avoir à choisir les caractéristiques à imposer fait qu'on peut avoir plusieurs méthodes dans cette approche. La plus efficace d'entre elles reste celle de Yang et Huang. [1]



Figure 1.12 : Le visage typique de Yang et Huang.[1]

IV.4.3.2 Méthodes basées sur les caractéristiques invariantes

Un des inconvénients des méthodes de l'approche précédente est qu'elles sont affectées par les conditions d'éclairage et d'illumination. Les méthodes de la présente approche visent à trouver les caractéristiques invariantes d'un visage qui ne peuvent pas être affectées par des facteurs externes.

L'idée de cette approche vient du fait que l'être humain peut détecter un visage quelles que soient les conditions d'éclairage, la taille ou la position du visage dans l'image. Cela montre l'existence de caractéristiques fixes qui ne changent jamais dans un visage. Parmi ces caractéristiques, figure en particulier la couleur de la peau.

En effet la couleur de la peau d'un être humain a une caractéristique importante et particulière : la distribution de la couleur de la peau humaine ne change pas par rapport à la chrominance. Grâce à cela, on peut détecter la couleur de la peau sur une image donc on peut détecter des régions de l'image qui ont la couleur de la peau humaine.

On peut citer d'autres caractéristiques faciales invariables, par exemple si la taille d'une bouche varie d'une personne à une autre, elle a toujours la même forme. On peut extraire un rapport sur la bouche qui sera fixe quelle que soit la taille de cette bouche. Généralement on utilise une combinaison de caractéristiques invariables pour résoudre le problème de la détection. [1]



Figure 1.13 : Détection de la couleur de la peau dans diverses images.[1]

IV.4.3.3 Méthodes basées sur la mise en correspondance

Dans cette approche, on crée quelques patterns normatifs (visages moyens) pour décrire un visage, puis on calcule la corrélation entre l'image d'entrée et le pattern.

Grâce à cela on dépiste des visages dans une image.

Dans cette méthode il y a deux facteurs importants :

- La façon de construction du pattern du visage et l'algorithme de correspondance.
- Les patterns de visages représentent l'ensemble des visages, nous les obtenons manuellement. On extrait un ensemble de visages aux quels nous imposons plusieurs conditions pour être bien choisis, puis on définit le pattern comme étant la moyenne de tous les visages. Pour renforcer le pattern on passe par quelques prétraitements comme l'égalisation de l'histogramme et l'élimination du fond.

La mise en correspondance est basée sur des fonctions de corrélation croisées de deux dimensions entre le pattern du visage en niveau de gris et une fenêtre de l'image en entrée en niveau de gris. Des patterns de visage peuvent être calculés indépendamment pour les yeux la bouche et autres.

Luhang Liang, Haizhou Ai, et Guangyou Xu [7] appliquent d'abord un pattern des yeux pour déterminer les régions pouvant contenir un visage puis un pattern de visage est appliqué à ces régions.



Figure 1.14 : a) Pattern de visage avec égalisation d'histogramme.
b) Pattern de visage après élimination du fond.[7]

V. Comparaison des méthodes de détection de visages

On a ici un tableau récapitulatif des avantages et inconvénients des différentes méthodes pour avoir une idée globale sur leur efficacité.

| Méthode | Avantages | Inconvénients |
|-------------------------------------|---|--|
| Template matching | <ul style="list-style-type: none"> - Conceptuellement simple - Mesure de similarités | <ul style="list-style-type: none"> - Recherche multi-échelle - Filtrage des multiples détections - Faible précision - Modèle représentatif |
| Eigen Object Eigen faces | <ul style="list-style-type: none"> - Gabarit moins crucial - Moins sensible à l'éclairage - Mesure de similarité | -Idem au Template matching |
| Réseaux de Neurones | <ul style="list-style-type: none"> - Apprentissage automatique - Capacité de généralisation | <ul style="list-style-type: none"> - Faible précision - Recherche multi-échelle |
| Support vector machine | -simple à mettre en œuvre | <ul style="list-style-type: none"> -faible performance -gourmande en temps et en précision. |

Tableau 1.1: Comparaison des méthodes de détection de visages.

VI. Suivi du visage dans une vidéo numérique

VI.1 La vidéo

Une vidéo est une succession d'images à une certaine cadence. L'œil humain a comme caractéristique d'être capable de distinguer environ 20 images par seconde. Ainsi, en affichant plus de 20 images par seconde, il est possible de tromper l'œil et de lui faire croire à une image animée. On caractérise la fluidité d'une vidéo par le nombre d'images par secondes (en anglais *frame rate*), exprimé en *FPS* (*Frames per second*, en français *trames par seconde*) [10].

VI.2 Formats vidéo

Il existe plusieurs formats de fichiers vidéo, à l'intérieur desquels le choix de la qualité est inversement proportionnel à la taille du fichier généré. Les formats les plus courants sont :

VI.2.1 AVI (Audio Video Interleave)

Format de fichier d'animations mis au point par Microsoft [11], qui fonctionne sur n'importe quelle machine, et ne nécessite pas de carte d'extension particulière. Dans ce format, la compression est toujours effectuée image par image (12 Go pour 10 mn de vidéo).

VI.2.2 MPEG (Motion Picture Expert Group)

Le groupe MPEG a été établi en 1988 dans le but de développer des standards internationaux de compression, décompression, traitement et codage d'image animée et de données audio. Dans la plupart des séquences vidéos, la majorité des scènes sont fixes ou bien changent très peu.

Par exemple lorsque seules les lèvres de l'acteur bougent, presque seuls les pixels de la bouche vont être modifiés d'une image à l'autre, il suffit donc de ne décrire seulement le changement d'une image à l'autre.

Il existe plusieurs standards MPEG [12]:

- **MPEG-1** : développé en 1988, est un standard pour la compression des données

vidéo. Il permet le stockage de vidéos à un débit de 1.5Mbps.

- **MPEG-2** : un standard dédié originalement à la télévision numérique (HDTV) offrant une qualité élevée à un débit pouvant aller jusqu'à 40 Mbps. Le MPEG-2 permet de plus une identification et une protection contre le piratage. Il s'agit du format utilisé par les DVD vidéo.
- **MPEG-4** : un standard destiné à permettre le codage de données multimédia sous formes d'objets numériques, afin d'obtenir une plus grande interactivité, ce qui rend son usage particulièrement adapté au Web et aux périphériques mobiles.

VI.2.3 Windows Media (ASF, WMV)

Format audio/vidéo développé pour Internet par Microsoft. Il utilise la norme MPEG-4 [13] :

- Advanced Streaming Format (ASF): basé sur la norme MPEG-4 V2.
- Windows Media Vidéo (WMV) : basé sur la norme MPEG-4 V3 (plus moderne).

| Format | Résolution | Type de compression | Débit | Applications |
|--------|------------|--------------------------|------------------|---|
| MPEG | 720 x 486 | Intratrame | 0,5 – 25 Mo/s | Générales |
| MPEG-1 | 352 x 240 | Intratrame | 0,01 – 0,06 Mo/s | CD-ROM, Internet |
| MPEG-2 | 720 x 480 | Intratrame et intertrame | 0,01 - 2 Mo/s | DVD, TV satellite |
| DV | 720 x 480 | Intratrame | 3,5 Mo/s | Grand public, professionnelles, télédiffusion |
| D1 | 720 x 486 | aucun | 25 Mo/s | Télédiffusion |

Figure 1.15 : Exemples de codes vidéo avec domaines d'application.

Le débit nécessaire pour l'affichage d'une vidéo, c'est le nombre d'octets affichés (ou transférés) par unité de temps.

$$\text{Débit} = \text{Taille de l'image} \times \text{Nombre d'images par seconde}$$

VI.3 La détection des points d'intérêt

Afin de suivre le visage dans toute la séquence, il faut dans un premier temps le détecter dans la première image de la séquence. L'objectif étant ici de suivre quelque point particulier du visage, par exemple par l'algorithme KLT, pour détecter les yeux dans une image couleur. Cette méthode nous permet d'obtenir la position des yeux et donc celle du visage dans la

première image de la séquence. On considère également pour le suivi, la position du nez sur le visage. Celle-ci peut être facilement obtenue à partir des positions des deux yeux en utilisant la distance interoculaire et la configuration géométrique des éléments du visage

VI.4 Méthodes de suivi des points d'intérêt

Il existe de nombreuses méthodes de suivi des points d'intérêt, les techniques classiques de suivi rapide du visage sont basées principalement sur l'utilisation de la couleur de celui-ci, d'autres méthodes sont basées sur les caractéristiques faciales (œil, nez, bouche), les méthodes les plus utilisées sont les deux méthodes : la méthode KLT et la méthode "block-matching". Une fois les points d'intérêt détectés dans la première image de la séquence, leur position est estimée dans les images suivantes en utilisant une méthode de suivi.



Figure 1.16 : Exemple de détection des points d'intérêt. De gauche à droite : première image de la séquence ; les zones d'intérêt, yeux et nez, détectées de manière automatique.

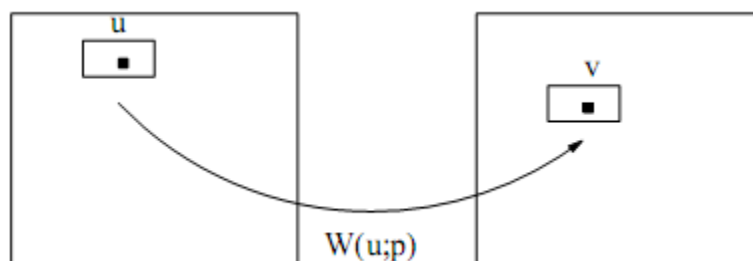


Figure 1.17 : Principe de la méthode KLT. On recherche la transformation W qui minimise la somme des erreurs quadratiques

VI.4.1 L'algorithme KLT

L'algorithme KLT est une méthode largement utilisée en vision par ordinateur aussi bien pour l'alignement ou le recalage d'images, que pour le suivi de régions d'intérêt dans une séquence d'images. Il tire son nom des initiales de ses auteurs. La première version a été proposée en 1981 par Lucas et Kanade dans le cadre du recalage d'images. Elle a ensuite été développée par Tomasi et Kanade, d'où le nom KLT pour Kanade-Lucas-Tomasi. Une description très complète de l'algorithme KLT est présentée par Baker et Matthews dans [9].

le principe de la méthode est le suivant : Soient I et J deux images, et $u = (ux, uy)^T$ un point de l'image I . On souhaite trouver le point v de J tel que les quantités $I(u)$ et $J(v)$ soient "similaires". On notera que cela s'apparente au problème de la mise en correspondance d'image, et l'algorithme KLT est d'ailleurs souvent utilisé pour cette application.

VI.4.2 L'algorithme "block-matching"

L'algorithme "block-matching" est utilisé par Spors et Rabenstein pour le suivi des yeux, consiste à rechercher pour chaque point d'intérêt de l'image I_t , le point de l'image I_{t+1} qui maximise un score de corrélation. La recherche est effectuée dans une fenêtre d'intégration Ω . On commence par détecter les positions des points d'intérêt dans la première image de la séquence, et pour chaque point on extrait un masque de corrélation qui servira de référence.

VII. Conclusion

Dans ce chapitre nous avons fait un tour d'horizon sur les méthodes et les techniques existantes de la détection de visages sur des images fixes et dynamiques ou nous avons constaté que, malgré la diversité des méthodes qui traitent cette tâche, le problème n'est certes pas encore résolu.

- I. Introduction
- II. La perception de la couleur et la théorie trichromatique
- III. Détection de peau dans une image couleur
- IV. Conclusion

I. Introduction

La couleur est l'une des plus importantes informations dans le domaine du traitement des images numériques, qui peut être considéré comme vecteur à trois dimensions définis selon l'espace colorimétrique utilisée.

Le choix de l'espace colorimétrique est une opération très délicate car il influence sensiblement les résultats.

II. La perception de la couleur et la théorie trichromatique

La compréhension de la perception de la couleur a commencé avec Newton et son expérience sur la dispersion de la lumière à travers un prisme en 1672. En 1801, Thomas Young a suggéré que trois couleurs primaires étaient suffisantes pour produire toutes les couleurs de façon additive. Ces travaux ont été poursuivis par Helmholtz et la théorie trichromatique a été prouvée en 1960 par la découverte de trois types de récepteurs dans la rétine qui correspondent aux trois types de cônes L, M et S. Leur réponse maximale se situe respectivement dans les teintes bleues à 440 nm pour les cônes de type S (Short) dans les teintes vertes à 545 nm pour les cônes de types M (Medium) et dans les teintes rouges à 580 nm pour les cônes de type L (Long). Du fait de cette trichromie, il est possible de représenter les couleurs dans un espace tridimensionnels dont les vecteurs de base correspondent aux couleurs primaires. Ainsi, un vecteur couleur [S] est défini par une combinaison linéaire des vecteurs de la base ([R], [G], [B]) :

$$[S] = r[R] + g[G] + b[B]$$

Où les nombres r, g et b sont les composantes trichromatiques et représentent les quantités

respectives des primaires utilisées. D'importants travaux ont été effectués afin d'obtenir les fonctions colorimétriques qui permettent de calculer facilement les composantes trichromatiques d'une lumière colorée. Ces travaux ont donné naissance au standard défini par la CIE (Commission Internationale de l'Eclairage) en 1931 [14].

II.1 Espaces de la couleur

En référence au système visuel humain et à la théorie trichromatique, on considère, de manière générale, que la couleur se définit selon trois composantes qui conduisent à différentes familles de systèmes de représentation : les systèmes primaires, les systèmes luminance-chrominance [15].

II.1.1 Systèmes luminance-chrominance

L'intérêt des espaces de type luminance-chrominance est qu'ils dissocient la composante de luminance des composantes de chrominance. De nombreux espaces de représentation se rattachent à cette famille et ils se différencient essentiellement par la façon dont sont calculées les coordonnées de luminance et de chrominance.

II.1.1.1 Espaces de type YCrCb

Ce système a été à l'origine développé pour assurer une compatibilité entre les téléviseurs couleurs et les téléviseurs noir et blanc, d'où la séparation des composantes de luminance et de chrominance. Une simple transformation linéaire permet de passer d'un système RGB au système YCrCb, mais cette transformation diffère suivant les standards de télévision (NTSC, PAL ou SECAM).

La forme générale des composantes chromatiques est donnée par :

$$Cb = a1(R - Y) + b1 (B - Y)$$

$$Cr = a2(R - Y) + b2 (B - Y)$$

Où les coefficients a_1 , a_2 , b_1 et b_2 sont spécifiques au standard considéré et Y est la luminance. Comme déjà souligné, il existe plusieurs systèmes de type YCrCb.

a) système YCbCr

Le système YCbCr est utilisé pour les images JPEG. Ce modèle colorimétrique permet en effet de réduire la taille d'une image. Cette réduction se base sur la constatation suivante : l'oeil humain est plus sensible à la luminance qu'à la chrominance.

$$\begin{pmatrix} Y \\ C_b \\ C_r \end{pmatrix} = \begin{pmatrix} 0,2220 & 0,7067 & 0,0713 \\ -0,1195 & -0,3810 & 0,5000 \\ 0,5000 & -0,4542 & -0,0458 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix}$$

b) système YIQ

Le système YIQ est celui qui correspond à La norme TV NTSC des États-Unis utilise l'encodage Y I Q :

$$\begin{pmatrix} Y \\ I \\ Q \end{pmatrix} = \begin{pmatrix} 0.30 & 0.59 & 0.11 \\ 0.60 & -0.28 & -0.32 \\ 0.21 & -0.52 & -0.31 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix}$$

Approximation de :

Y = Luminance
I = rouge - cyan
Q = magenta - vert

Les téléviseurs noir et blanc n'affiche que Y.

c) système YUV

Le signal YUV est créé depuis une source RGB (rouge, vert et bleu). Le signal U est obtenu en soustrayant le Y du signal bleu d'origine ; de façon similaire le V est obtenu en soustrayant Y du signal rouge. Ces opérations peuvent facilement être réalisées au moyen d'un circuit analogique [16]. Les équations suivantes peuvent être utilisées pour dériver Y, U et V à partir des composantes R, G et B :

$$(R, G, B, Y) \in [0 ; 1]$$

$$U \in [-0,436 ; 0,436]$$

$$V \in [-0,615 ; 0,615]$$

Cependant, les limites autorisées sur U et V dépendent de Y .

De RVB à YUV :

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0,299 & 0,587 & 0,114 \\ -0,14713 & -0,28886 & 0,436 \\ 0,615 & -0,51498 & -0,10001 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

II.1.2 Les systèmes perceptuels

Les systèmes perceptuels correspondent aux systèmes uniformes au sens de la perception visuelle. Plus précisément, la spécificité de ces systèmes est de pouvoir décrire fidèlement les écarts de couleur entre couleurs proches. Ils sont donc dits, perceptuellement uniformes [15].

II.1.2.1 L'espace L*a*b*

Le système L*a*b* peut être considéré comme le système de référence de la CIE (il a été introduit en 1976) pour calculer des écarts de couleurs. La transformation qui permet de passer de l'espace XYZ (l'espace XYZ est un espace de primaires introduit par la CIE en 1931 pour pallier à certains inconvénients du système colorimétriques RGB) à l'espace L*a*b* est une transformation non linéaire qui prend en compte les coordonnées trichromatiques du blanc de référence $W = (X_0, Y_0, Z_0)$ [15].

Les composantes L*, a* et b* sont obtenues par les équations suivantes :

$$L^* = \begin{cases} 116 * (\frac{Y}{Y_0})^{\frac{1}{3}} - 16 & \text{si } \frac{Y}{Y_0} \geq 0.008856 \\ 903.3 * \frac{Y}{Y_0} & \text{si } \frac{Y}{Y_0} \leq 0.008856 \end{cases}$$

$$a^* = 500 * (f(\frac{X}{X_0}) - f(\frac{Y}{Y_0}))$$

$$b^* = 200 * (f(\frac{Y}{Y_0}) - f(\frac{Z}{Z_0}))$$

$$f(x) = \begin{cases} x^{\frac{1}{3}} & \text{si } \frac{Y}{Y_0} \geq 0.008856 \\ 7.787x + \frac{16}{116} & \text{si } \frac{Y}{Y_0} \leq 0.008856 \end{cases}$$

II.1.2.2 L'espace HSI

La modélisation de ce système de représentation communément utilisé en traitement d'images couleur, est issue de la rotation du cube des couleurs RGB. En effet, il suffit de faire pivoter le cube sur le coin représentant le noir ; ainsi, l'axe achromatique constitue l'axe des intensités I et la couleur est dénie par une position sur un pallier circulaire où la saturation S représente le rayon et la teinte H représente l'angle. Les formules exprimant la transformation de l'espace RGB à l'espace HSI sont données par :

$$\begin{cases} I = \frac{R+G+B}{3} \\ S = 1 - \frac{3 * \min(R,G,B)}{R+G+B} \\ H = \arccos\left(\frac{0.5 * (R-G) + (R-B)}{\sqrt{(R-G)^2 + (R-B)(G-B)}}\right) \end{cases}$$

II.1.2.3 L'espace HSV

L'espace HSV est un système de cône hexagonal qui représente la couleur sous la forme d'un triplé : teinte H (Hue), saturation S (Saturation) et luminosité V (Value). Les transformations sont effectuées comme suit [15]:

$$V = \max(R, G, B)$$

$$S = \frac{V - \min(R, G, B)}{V}$$

$$H = \begin{cases} \frac{G-B}{V-\min(R,G,B)} & \text{si } V = R \\ 2 + \frac{B-R}{V-\min(R,G,B)} & \text{si } V = G \\ 4 + \frac{R-G}{V-\min(R,G,B)} & \text{si } V = B \end{cases}$$

III. Détection de peau dans une image couleur

La détection de peau est une étape très importante dans de nombreux systèmes de vision qui ont pour but la détection, la localisation et la reconnaissance de personnes. La détection de la peau est en effet, un des moyens efficace pour détecter une personne dans une image car, d'une manière générale, au moins quelques parties du corps telles que la tête et le visage, les bras et les mains sont visibles. La construction d'un système de détection de la peau doit cependant répondre à deux principaux problèmes [15] :

- Quel espace de représentation des couleurs choisir ?
- Comment modéliser la distribution de la couleur de la peau dans cet espace ?

III.1 Choix de l'espace couleur

Il existe de nombreux espaces de représentation de la couleur et le choix d'un espace dépend en priorité de l'application envisagée. Dans le cas de la détection de la peau, ce choix peut être guidé par deux observations [15] :

- Bien que la couleur de la peau varie d'une personne à une autre, différentes études ont montré que la différence se situe plus au niveau de la composante de luminance qu'au niveau des composantes de chrominance.
- Dans l'espace défini par les composantes de chrominance, les pixels correspondant à la peau sont "assez bien" regroupés.

Ces deux observations conduisent à privilégier les espaces de représentation qui séparent les composantes de luminance et de chrominance et à s'intéresser aux composantes de

chrominance. De nombreux auteurs utilisent les composantes de chrominance pour la détection de la peau dans différentes applications.

III.2 Modélisation de la peau

III.2.1 Distribution dans l'espace de chrominance

Afin de déterminer la distribution des composantes chromatiques caractérisant la peau dans les espaces RGB, YCrCb et HSI, des études ont été menées à noter que les pixels correspondants à la peau sont assez bien regroupés dans le plan des chrominances. On peut donc segmenter une image en deux régions en classant chaque pixel dans une des deux catégories peau et non-peau. Pour ce faire, il faut modéliser la zone de l'espace du plan de chrominances qui caractérise les pixels de peau et plusieurs approches sont possibles [15] :

- Les approches non-paramétriques qui visent à estimer la distribution de la couleur de la peau sans une modélisation explicite à partir d'un ensemble d'apprentissage. Elles se traduisent généralement par un histogramme et une carte de probabilité SPM (Skin ProbabilityMap) qui affecte une valeur de probabilité à chaque pixel de l'espace discrétisé.
- Les approches paramétriques qui estiment la distribution de la couleur de la peau sous forme de modèles explicites. La distribution est généralement caractérisée par une densité ou une somme de densités de probabilité dont les paramètres sont obtenus à partir d'un ensemble d'apprentissage.

Les approches non-paramétriques possèdent l'avantage de la rapidité (à la fois dans les phases d'apprentissage et de classification) et sont, en théorie, indépendantes de la forme de la distribution. Elles nécessitent cependant un espace de stockage important (pour représenter la carte de probabilité) ainsi qu'un ensemble d'apprentissage de grande taille. La phase d'apprentissage peut être plus longue avec les approches paramétriques, mais ces dernières possèdent l'avantage de fournir une bonne estimation de la densité avec un ensemble d'apprentissage plus réduit. Les deux approches sont largement employées dans la littérature et donnent des résultats comparables.

Les modèles les plus couramment utilisés sont le modèle gaussien qui représente cette distribution sous la forme d'une gaussienne, et le modèle de mélange de gaussiennes (Mixture

of Gaussians) qui représente la distribution sous la forme d'une somme de gaussiennes. Ce dernier modèle est semble-t-il plus adapté pour prendre en compte la variabilité des conditions d'acquisition des images et la présence de populations hétérogènes. La distribution d'une variable aléatoire $X \in \mathbb{R}^d$ est un mélange de k gaussiennes si sa fonction de densité est de la forme :

$$f(x|\theta) = \sum_{j=1}^k \alpha_j \frac{1}{\sqrt{(2\pi)^d |\Sigma_j|}} * \exp\left\{-\frac{1}{2}(x - \mu_j)^T \Sigma_j^{-1} (x - \mu_j)\right\}$$

f est donc la somme pondérée de k gaussiennes f_j , de moyennes et de matrices de covariance respectives μ_j et Σ_j . Les termes α_j sont des coefficients de pondération et vérifient la relation

$$\sum_{j=1}^k \alpha_j = 1$$

L'estimation de l'ensemble des paramètres $\theta = \{\alpha_j, \mu_j, \Sigma_j\}_{j=1}^k$ du modèle, à partir d'un ensemble d'apprentissage, peut se faire grâce à l'algorithme EM (Expectation Maximisation) bien connu en statistiques.

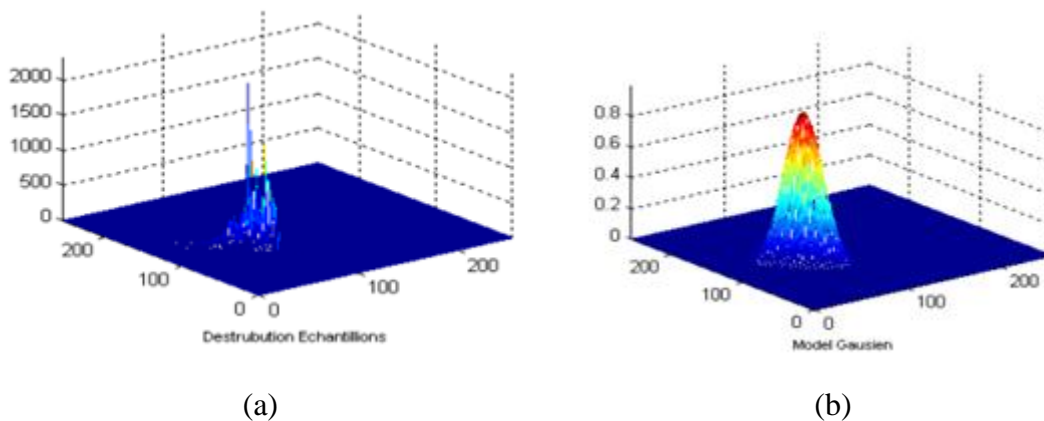


Figure 2.1 : (a) Histogramme des pixels de peau (b) modèle gaussien de peau dans l'espace RGB.

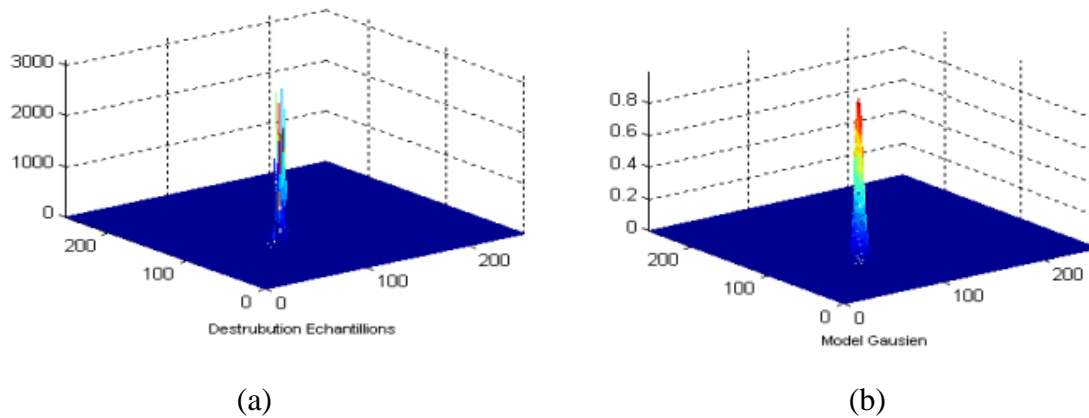


Figure 2.2 : (a) Histogramme des pixels de peau (b) modèle gaussien de peau dans l'espace $YCrCb$

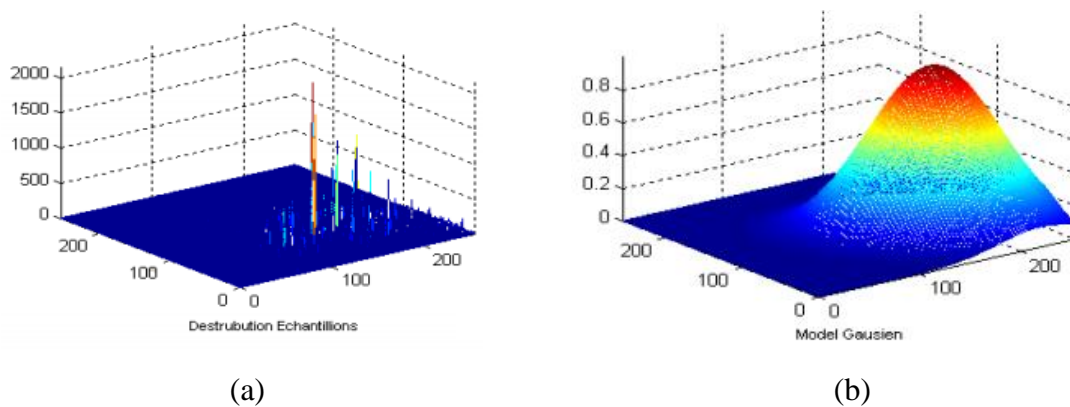


Figure 2.3 : (a) Histogramme des pixels de peau (b) modèle gaussien de peau dans l'espace HSI .

IV. Conclusion

Dans ce chapitre, nous avons vu comment l'information colorimétrique présente dans une image couleur peut être utilisée pour la détection de la peau. Cela passe par le choix d'un espace de représentation convenable et par une modélisation adéquate de la distribution des couleurs dans cet espace. Ainsi que espaces de type luminance chrominance sont les plus adapter pour la présentation de la peau.

- I. Introduction
- II. Historique
- III. Avantage par rapport aux méthodes classiques
- IV. Réseaux de neurone
- V. Le perceptron multicouche
- VI. Domaine d'applications
- VII. Conclusion

I. Introduction :

Centaines exemples nous montrent à la fois combien on peut espérer de la modélisation du système nerveux mais aussi combien il sera difficile d'imaginer et de comprendre les divers aspects des problèmes de perception. Il paraît donc naturel d'essayer de comprendre comment les systèmes biologiques sont capables de telles performances, et si possible, de s'inspirer de leurs principes pour imaginer de nouveaux algorithmes ou de nouvelles machines plus efficaces que ceux dont nous disposons actuellement.

Les techniques de réseaux de neurones relèvent d'une telle approche :

Comprendre les principes selon lesquels les systèmes biologiques traitent l'information et s'en inspirer pour élaborer de nouvelles techniques en sciences de l'ingénieur. C'est donc une double démarche, à la fois cognitive et synthétique où le monde biologique doit être considéré comme une source de référence et de connaissance. [17]

II. Historique :

Le champ des réseaux neuronaux va démarrer par la présentation **en 1943** par W. McCulloch et W. Pitts [17] du neurone formel qui est une abstraction du neurone physiologique. Le retentissement va être énorme. Par cette présentation, ils veulent démontrer que le cerveau est équivalent à une machine de Turing, la pensée devient alors purement des mécanismes matériels et logiques.

En 1949, D. Hebb présente dans son ouvrage « The Organization of Behavior » [17] une règle d'apprentissage. De nombreux modèles de réseaux aujourd'hui s'inspirent encore de la règle de Hebb.

En 1958, F. Rosenblatt développe le modèle du perceptron. C'est un réseau de neurones inspiré du système visuel. Il possède deux couches de neurones : une couche de perception et une couche liée à la prise de décision. C'est le premier système artificiel capable d'apprendre par expérience.

Dans la même période, Le modèle de L'Adaline (ADAPtiveLINarElement) a été présenté par B. Widrow et Hoff. Ce modèle sera par la suite le modèle de base des réseaux multi-couches.

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

C'est **en 1982** que J. Hopfield présente son étude d'un réseau complètement rebouclé, dont il analyse la dynamique. [17]

III. Avantage par rapport aux méthodes classiques

Malgré la constante augmentation de puissance des calculateurs, malgré les approches théoriques de plus en plus sophistiquées, un certain nombre de tâches résistent encore aux algorithmes et aux méthodes classiques de traitement des signaux et des données. Ces tâches relèvent typiquement du traitement en temps réel, de très grands flots de données souvent multidimensionnelle et arrivant à des cadences élevées. Le grand nombre des données, leur variabilité, le fait qu'elles ne répondent pas à des modèles physiques connus nous laissent souvent démunis devant des tâches de caractérisation, de reconnaissance et de prise de décision, dans cela les réseaux de neurones viennent de démontrer qu'ils sont plus fiables et plus puissants.[17]

IV. Réseaux de neurone

IV.1 Le neurone biologique

IV.1.1 Présentation :

Le système nerveux compte plus de 1000 milliards de neurones interconnectés. Bien que les neurones ne soient pas tous identiques, leur forme et certaines caractéristiques permettent de les répartir en quelques grandes classes. En effet, les neurones n'ont pas tous un comportement similaire en fonction de leur position dans le cerveau. [Figure 3.1]

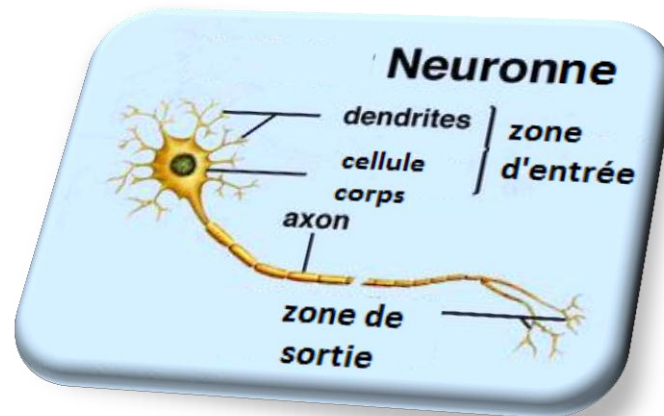


Figure 3.1 : neurone biologique

Le neurone se décompose en trois régions principales

IV.1.2 Le corps cellulaire :

Il contient le noyau du neurone ainsi que la machine biochimique nécessaire à la synthèse des enzymes. Ce corps cellulaire de forme sphérique ou pyramidale contient aussi autres molécules essentielles à la vie de la cellule. Sa taille est de quelques microns de diamètre. [18]

IV.1.3 Les dendrites :

Ce sont de fines extensions tubulaires qui se ramifient autour du neurone et forment une sorte de vaste arborescence. Les signaux envoyés au neurone sont captés par les dendrites. Leur taille est de quelques dizaines de microns de longueur. [18]

IV.1.4 L'axone:

C'est le long de l'axone que les signaux partent du neurone. Contrairement aux dendrites qui se ramifient autour du neurone, l'axone est plus long et se ramifie à son extrémité ou il se connecte aux dendrites des autres neurones. Sa taille peut varier entre quelques millimètres à plusieurs mètres. [18]

IV.1.5 Synapse :

Une synapse est une jonction entre deux neurones, et généralement entre l'axone d'un neurone et une dendrite d'un autre neurone. [18]

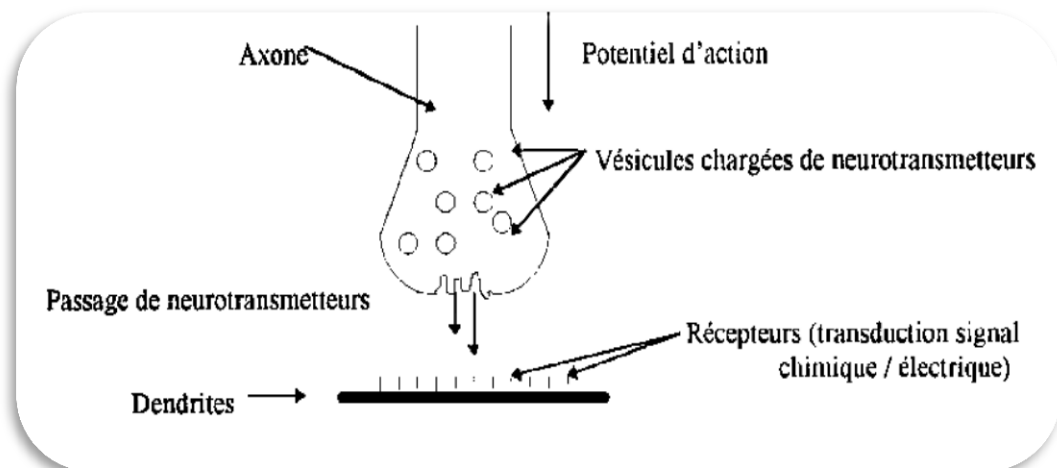


Figure 3.2 : Fonctionnement de neurone biologique [18]

➤ Fonctionnement

Le neurone reçoit des informations, venant d'autres neurones, grâce à ses dendrites. Il va ensuite y avoir sommation, au niveau du corps cellulaire, de toutes ces informations et via un potentiel d'action (un signal électrique) le résultat de l'analyse va transiter le long de l'axone jusqu'aux terminaisons synaptiques. A cet endroit, lors de l'arrivée du signal, des vésicules synaptiques vont venir fusionner avec la membrane cellulaire, ce qui va permettre la libération des neurotransmetteurs (médiateurs chimiques) dans la fente synaptique. Le signal électrique ne pouvant pas passer la synapse (dans le cas d'une synapse chimique), les neurotransmetteurs permettent donc le passage des informations, d'un neurone à un autre.

Au niveau post-synaptique, sur la membrane dendritique, se trouvent des récepteurs pour les neurotransmetteurs. Suivant le type de neurotransmetteur et le type des récepteurs, l'excitabilité du neurone suivant va augmenter ou diminuer, ce qui fera propager ou non l'information. [18]

IV.2 Le neurone Formel

IV.2.1 Présentation

Le neurone formel est une modélisation mathématique qui reprend aux principes de fonctionnement du neurone biologique, en particulier la sommation des entrées. Sachant qu'au niveau biologique, les synapses n'ont pas toutes la même «valeur» (les connexions entre les neurones étant plus ou moins fortes), les auteurs ont donc créé un algorithme qui pondère la somme de ses entrées par des poids synaptiques (coefficients de pondération). De plus, les 1 et les -1 en entrée sont là pour figurer une synapse excitatrice ou inhibitrice.

les RNA possède certaine caractéristique notons : les poids, les biais, la sortie, la somation, la fonction de transfert. [19]

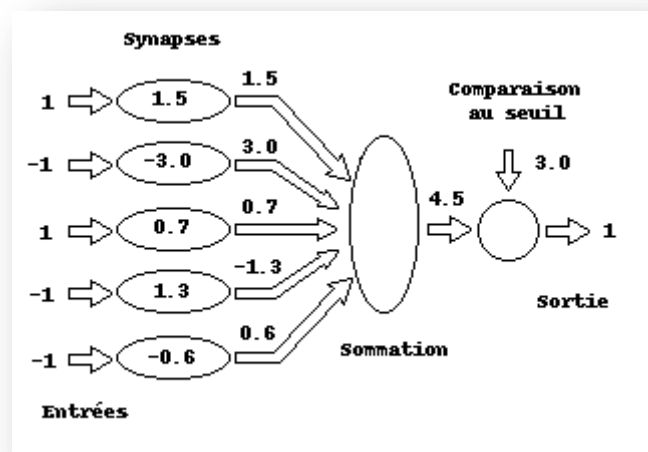


Figure 3.3 : Réseau de neurones artificiels [19]

IV.2.2 Caractéristiques du RNA

IV.2.2.1 Sommutation :

Pour un nombre compris entre $j(=1)$ et un nombre quelconque R , le neurone formel va calculer la somme de ses entrées (x_1, \dots, x_n), pondérées par les poids synaptiques (w_1, \dots, w_n). [17]

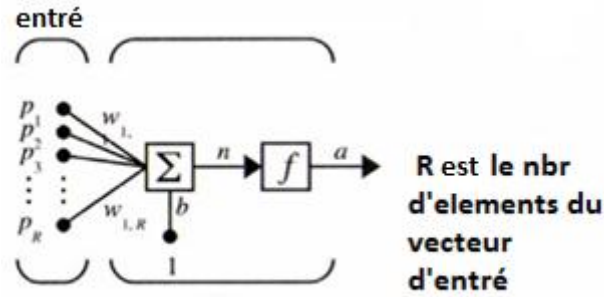


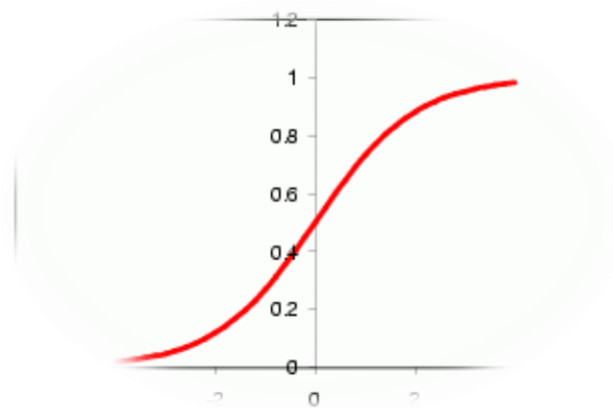
Figure 3.4 sommation du RNA [17]

Entrée du réseau : $n = w_{1,1}p_1 + w_{1,2}p_2 + \dots + w_{1,R}p_R + b$.

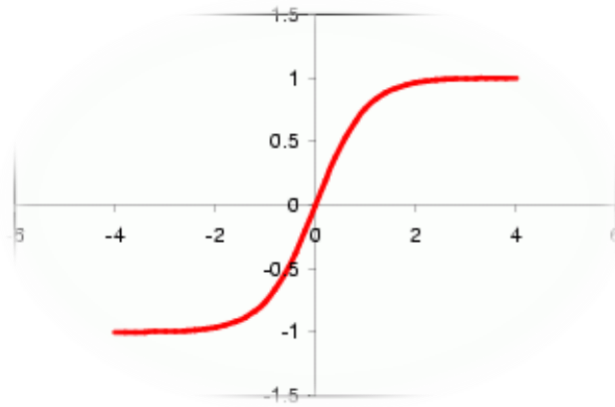
IV.2.2.2 La fonction de transfert [20] :

Différentes fonctions de transfert pouvant être utilisées comme fonction d'activation du neurone. les fonctions les plus utilisées sont :

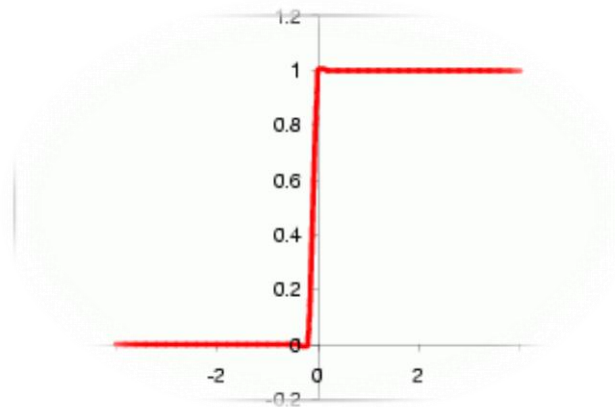
➤ **La fonction logistique:** $Y = F(X) = 1 / (1 + \exp(-d * X))$



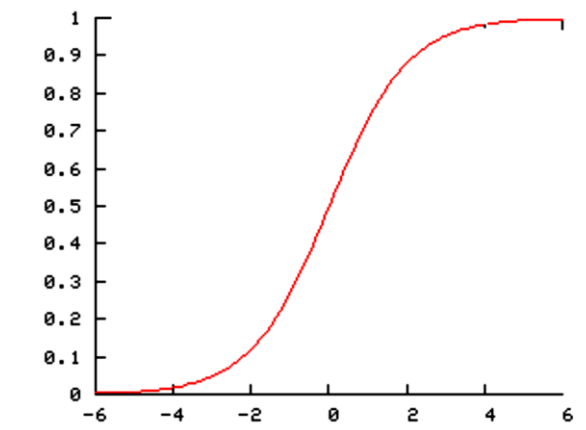
➤ **La tangente hyperbolique :** $Y = 2 / (1 + \exp(-2 * X)) - 1$



➤ La fonction Gaussienne : $Y = \exp(-(X^2)/2)$



➤ La fonction sigmoïde : $Y = 1/(1+\exp(-X))$



IV.2.2.3 Architecture :

Il existe différentes topologies de réseaux correspondant aux divers schémas géométriques de connexion entre les neurones : le réseau multicouche, le réseau à

connexions locales, le réseau à connexions récurrentes, le réseau à connexions complète.,, Les connexions entre les neurones qui composent le réseau décrivent la topologie du modèle. Elle peut être quelconque, mais le plus souvent il est possible de distinguer une certaine régularité. [21]

a) Réseau multicouche

Les neurones sont arrangés par couche, il n'y a pas de connexion entre neurones d'une même couche. Habituellement, chaque neurone d'une couche est connecté à tous les neurones de la couche suivante et celle-ci seulement, d'où la notion de sens de parcours de l'information au sein d'un réseau et la notion de neurones d'entrée et neurones de sortie. [21]

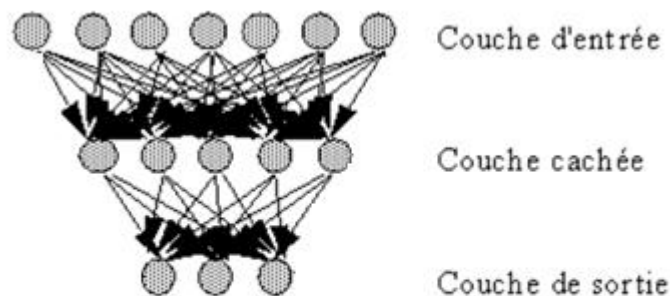


Figure 3.5 : Réseau multicouches.[21]

b) Réseau à connexions locales

„ Il s'agit d'une structure multicouche. Chaque neurone entretient des relations avec un nombre réduit et localisé de neurones de la couche avale, les connexions sont donc moins nombreuses que dans le cas d'un réseau multicouche classique. [19]

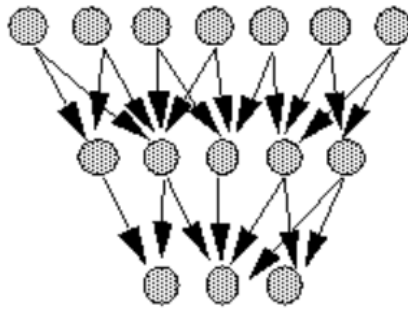


Figure 3.6 : Réseau à connexions locales[21].

c) Réseau à connexions récurrentes

„ Les connexions récurrentes ramènent l'information en arrière par rapport au sens de propagation défini dans un réseau multicouche. Ces connexions sont le plus souvent locales. [21]

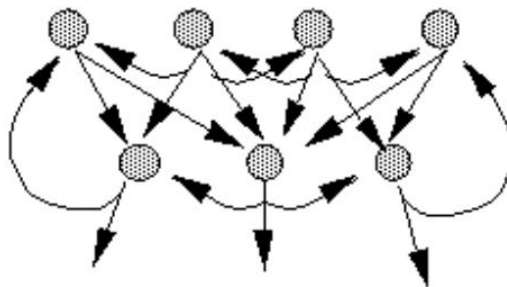


Figure 3.7 : Réseau à connexions récurrentes.

d) Réseau à connexion complète

„ C'est la structure d'interconnexion la plus générale. Chaque neurone est connecté à tous les neurones (et lui-même). [21]

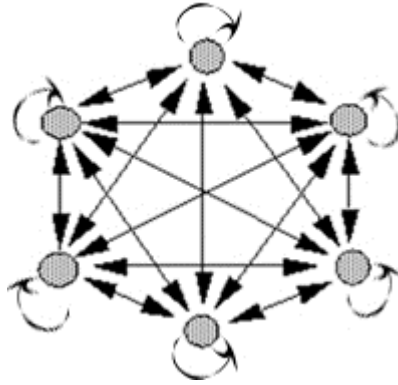


Figure 3.8: Réseau à connexion complète.

IV.2.3 Apprentissage :

L'apprentissage est une phase très importante pour le déploiement d'un réseau de neurone durant laquelle le comportement du réseau est modifié jusqu'à l'obtention du comportement désiré.

Généralement, les variables modifiées pendant l'apprentissage sont les poids des connexions. L'apprentissage adopté n'est, donc, que la modification des poids du réseau pour accorder sa réponse aux échantillons et à l'expérience. [22]

IV.2.3.1 Types d'apprentissage :

a) Le mode supervisé :

Dans ce type d'apprentissage, le réseau s'adapte par comparaison entre le résultat qu'il a calculé, en fonction des entrées fournies, et la réponse attendue en sortie. Ainsi, le réseau va se modifier jusqu'à ce qu'il trouve la bonne sortie, c'est-à-dire celle attendue, correspondant à une entrée donnée. [19]

b) Le renforcement :

Le renforcement est en fait une sorte d'apprentissage supervisé. Dans cette approche le réseau doit apprendre la corrélation entrée/sortie via une estimation de son erreur, c'est-à-dire du rapport échec/ succès. Le réseau va donc tendre à maximiser un index de performance qui lui est fourni, appelé signal de renforcement. Le système étant capable ici, de savoir si la

réponse qu'il fournit est correcte ou non, mais il ne connaît pas la bonne réponse. [19]

c) Le mode non-supervisé (ou auto-organisationnel) :

Dans ce cas, l'apprentissage est basé sur des probabilités. Le réseau va se modifier en fonction des régularités statistiques de l'entrée et établir des catégories, en attribuant et en optimisant une valeur de qualité, aux catégories reconnues. [19]

d) Le mode hybride :

Le mode hybride reprend en fait les deux autres approches, puisque une partie des poids va être déterminée par apprentissage supervisé et l'autre partie par apprentissage non-supervisé. [19]

IV.2.3.2 Règles d'apprentissage:

a) Règle de correction d'erreurs :

Cette règle s'inscrit dans le paradigme d'apprentissage supervisé, c'est -à-dire dans le cas où l'on fournit au réseau une entrée et la sortie correspondante. Si on considère « y » comme étant la sortie calculée par le réseau, et « d » la sortie désirée, le principe de cette règle est d'utiliser l'erreur (d-y), afin de modifier les connexions et de diminuer ainsi l'erreur globale du système. Le réseau va donc s'adapter jusqu'à ce que « y » soit égal à « d ». (Ce Principe est notamment utilisé dans le modèle du perceptron simple). [19]

b) Apprentissage de Boltzmann :

Les réseaux de Boltzmann sont des réseaux symétriques récurrents. Ils possèdent deux sous-groupes de cellules, le premier étant relié à l'environnement (cellules dites visibles) et le second ne l'étant pas (cellules dites cachées). Cette règle d'apprentissage est de type stochastique (qui relève partiellement du hasard) et elle consiste à ajuster les poids des connexions, de telle sorte que l'état des cellules visibles satisfasse une distribution probabiliste souhaitée. [19]

c) Règles de Hebb :

Cette règle est basée sur des données biologiques, modélise le fait que si des neurones, de part et d'autre d'une synapse, sont activés de façon synchrone et répétée, la force de la connexion synaptique va aller croissant. Il est à noter ici que l'apprentissage est localisé, c'est-à-dire que la modification d'un poids synaptique w_{ij} ne dépend que de l'activation d'un neurone i et d'un autre neurone j . [19]

d) Règle d'apprentissage par compétitions :

La particularité de cette règle, c'est qu'ici l'apprentissage ne concerne qu'un seul neurone. Le principe de cet apprentissage est de regrouper les données en catégories. Les patrons similaires vont donc être rangés dans une même classe, en se basant sur les corrélations des données, et seront représentés par un seul neurone, on parle de « winner-take-all ».

Dans un réseau à compétition simple, chaque neurone de sortie est connecté aux neurones de la couche d'entrée, aux autres cellules de la couche de sortie (connexions inhibitrices) et à elle-même (connexion excitatrice). La sortie va donc dépendre de la compétition entre les connexions inhibitrices et excitatrices. [19]

e) Règle du delta

La règle du delta, ou règle de Widrow-Hoff, utilise la **méthode des moindres carrés** qui mesure la divergence entre la configuration de sortie souhaitée et la configuration de sortie calculée

$$Q = \sum_i (c_i - s_i)^2$$

Q est la somme des carrés des différences entre la sortie calculée c_i et la sortie souhaitée s_i et sert de mesure de l'erreur qu'il s'agit de minimiser. On montre que la correction des poids, pour des fonctions d'activations sigmoïdes, est donnée par

$$dw_{ij} = a_i * (1 - a_i) * (c_i - a_i)$$

S'il y a une erreur, c'est à dire si $(c_i - a_i)$ n'est pas nul, et si a_i n'est pas nul, alors dw_{ij} n'est pas nul et il y a correction du poids w_{ij} , sinon le poids reste inchangé.[23]

V. Le perceptron multicouche

V.1 Définition

Le perceptron multicouche est un réseau orienté de neurones artificiels organisé en couches et où l'information voyage dans un seul sens, de la couche d'entrée vers la couche de sortie. La (figure 3.9) donne l'exemple d'un réseau contenant une couche d'entrée, deux couches cachées et une couche de sortie. La couche d'entrée représente toujours une couche virtuelle associée aux entrées du système. Elle ne contient aucun neurone. Les couches suivantes sont des couches de neurones. Dans l'exemple illustré, il y a 3 entrées, 4 neurones sur la première couche cachée, trois neurones sur la deuxième et quatre neurones sur la couche de sortie. Les sorties des neurones de la dernière couche correspondent toujours aux sorties du système. Dans le cas général, un perceptron multicouche peut posséder un nombre de couches quelconque et un nombre de neurones (ou d'entrées) par couche également quelconque.

Les neurones sont reliés entre eux par des connexions pondérées. Ce sont les poids de ces connexions qui gouvernent le fonctionnement du réseau et "programment" une application de l'espace des entrées vers l'espace des sorties à l'aide d'une transformation non linéaire. La création d'un perceptron multicouche pour résoudre un problème donné passe donc par l'inférence de la meilleure application possible telle que définie par un ensemble de données d'apprentissage constituées de paires de vecteurs d'entrées et de sorties désirées. Cette inférence peut se faire, entre autre, par l'algorithme dit de rétro propagation.

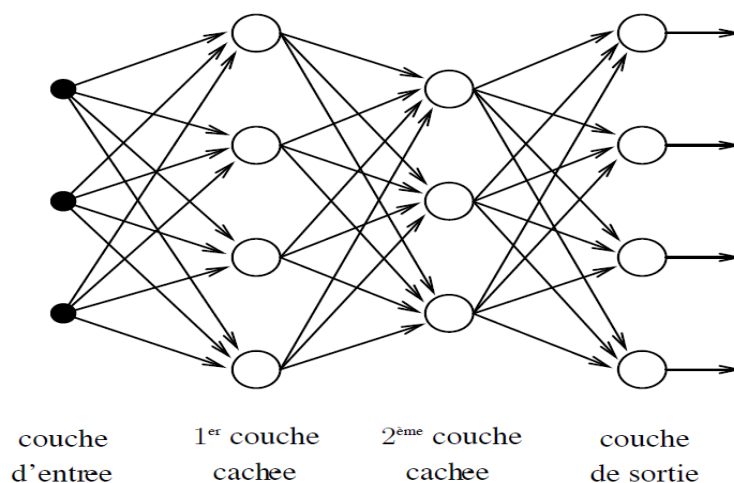


Figure 3.9 : Réseau de type perceptron multicouche.

V.2 Algorithme de rétro propagation

Cet algorithme nécessite une fonction continue, non-linéaire et différentiable comme fonction de transfert du neurone. L'algorithme de rétro propagation standard se résume donc à la série d'étapes suivantes [09]:

- 1/ Initialisation des poids à des valeurs aléatoires de faible grandeur;
- 2/ Sélection d'un exemple d'apprentissage (E, d) dans la base d'apprentissage
- 3/ Présentation de la forme d'entrée (E) sur la couche d'entrée du réseau;
- 4/ Calcul par propagation de la sortie obtenue (x);
- 5/ Si erreur en sortie alors pour tous les neurones i (depuis la sortie jusqu'à l'entrée)

Si i est un neurone de sortie alors $y_i = f(a_i) \cdot (1 - f(a_i)) \cdot (d_i - x_i)$

Si i est un neurone caché (ou d'entrée) alors $y_i = f(a_i) \cdot (1 - f(a_i)) \cdot \sum_k (w_{ki} \cdot y_k)$

(k : neurones compris entre la couche actuelle et la couche de sortie)

- 6/ Application de la procédure de gradient. μ est un gain fixé par l'utilisateur.

$$W_{ij}(t+1) = w_{ij}(t) + \mu \cdot y_i \cdot x_j$$

- 7/ Tant que l'erreur est trop importante, retour à l'étape 2

V.3 Propriétés des RNA

Les réseaux de neurones artificiels sont dotés d'un certain nombre de propriétés tel que [24]:

- Capacité de généralisation
- La rapidité
- La résistance au bruit
- La résistance à la dégradation
- L'adaptabilité

VI. Domaines d'applications :

Parmi les domaines d'application des RNA nous citons :

- Aérospatial : pilotage automatique, simulation du vol... etc.
- Automobile : système de guidage automatique,... etc.
- Défense : guidage de missile, suivi de cible, reconnaissance du visage, radar, sonar,

lidar, traitement du signal, compression de données, suppression du bruit...etc.

- Electronique : prédiction de la séquence d'un code, vision machine, synthétiseur vocal, modèle non linéaire,... etc.
- Finance : Prévion du coût de la vie
- Secteur médical : Analyse EEC et ECG
- Télécommunications : Compression de données ...etc. [17]

VII. Conclusion :

De nombreuses techniques sont actuellement explorées pour l'identification de visages. On peut citer à titre d'exemples l'utilisation de critères morphologiques, la détection de la couleur de la chair, ou la transformation en ondelettes. Celle que nous allons développer ici est la modélisation neuronale artificiel inspirer du neurone biologique.

Le grand avantage des réseaux de neurones réside dans leur capacité d'apprentissage automatique, ce qui permet de résoudre des problèmes sans nécessiter l'écriture de règles complexes tous en étant tolèrent aux erreurs.

VIII. Introduction

IX. Conception

X. Implémentation

XI. Base de données image

XII. Présentation de l'application

XIII. Test et résultat

XIV. Conclusion

I. Introduction

« **DFACE** » ou « Detecting Face » est un système de détection et suivi de visage en temps réel. Le fonctionnement du système « **DFACE** » est décomposé en deux principales phases :

- Extraction des segments peau où nous avons appliqué la mixture gaussienne.
- La détection et la localisation de visages par le réseau de neurones.

Le système « **DFACE** » nous offre la possibilité de détecter les visages sur une image fixe, une vidéo chargée à partir du disque dur, ou même en direct à partir d'une webcam.

II. Conception

Le schéma ci-dessous présente une description globale du système « **DFACE** »:

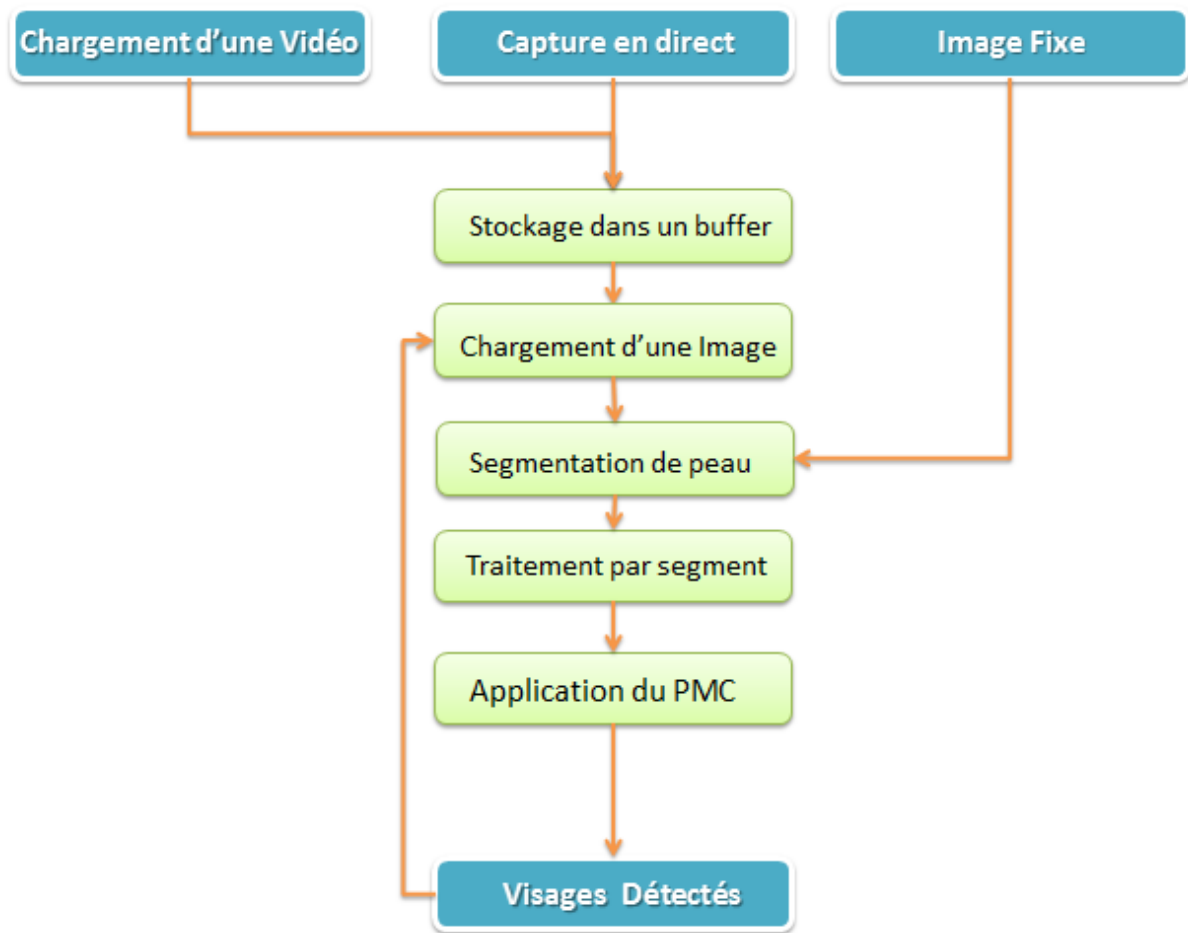


Figure 4.1 : Etapes de la conception

II.1 Stockage dans un buffer

Cette étape consiste à défragmenter en images une vidéo capturée d'une webcam ou chargée à partir du disque dur et les mettre dans un buffer.

II.2 Chargement d'une image

Les images stockées dans le buffer sont ordonnées par ordre d'apparition, il suffit de prendre chaque image et la traiter indépendamment.

II.3 Segmentation de peau

Plusieurs modèles de couleur de peau ont été proposés pour la détection des pixels de peau humains dans les images couleur. Par cela on peut distinguer 2 types de modèles :

- Modèle de peau non paramétrique :

La recherche selon cette approche vise à estimer la distribution de couleurs de peau des données d'apprentissage sans aucune hypothèse sur la distribution.

- Modèle de peau paramétrique :

Les modèles de peau paramétriques permettent d'ajuster les distributions avec quelques fonctions spécifiques paramétrées.

➤ Les méthodes paramétriques les plus utilisés pour construire un modèle de peau sont:

- le modèle basé sur une simple gaussienne.
- le modèle basé sur un mélange de gaussiennes.

La figure qui suit montre le processus de segmentation par couleur de peau :

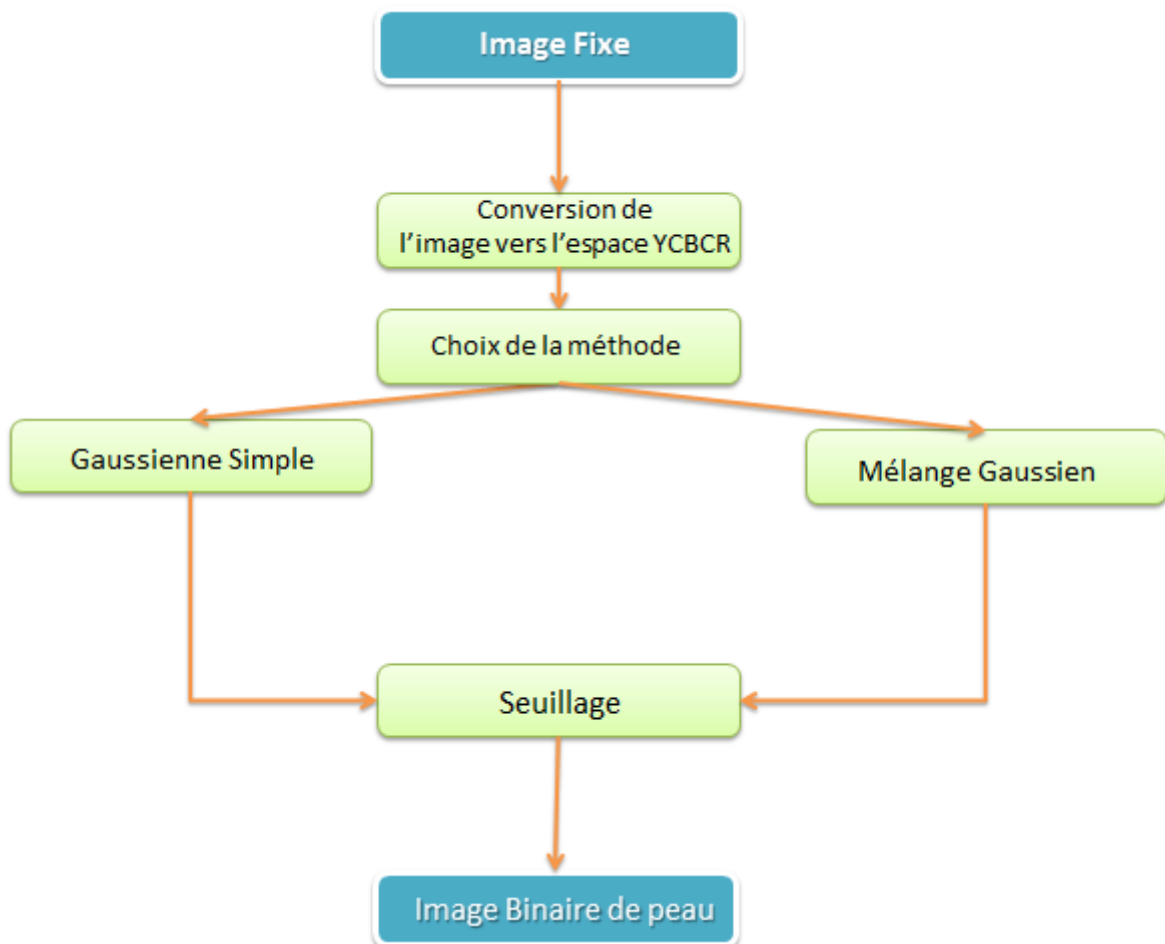


Figure 4.2 : Processus de détection de peau.

II.3.1 modèle basé sur une simple gaussienne

Le modèle gaussien est un modèle probabiliste de détection de peau qui s'appuie sur une base contenant les des trois types de couleur chair (blanche, brune, noir).

II.3.1.1 Création du model Gaussien simple

a) Préparation de la base d'échantillons

Les d'échantillons constituant la base peau sont issus de mains , fronts, joue ...etc. Tiré d'image de différentes sources : internet, vidéo, base caltetchetc. (Figure [4.3]).

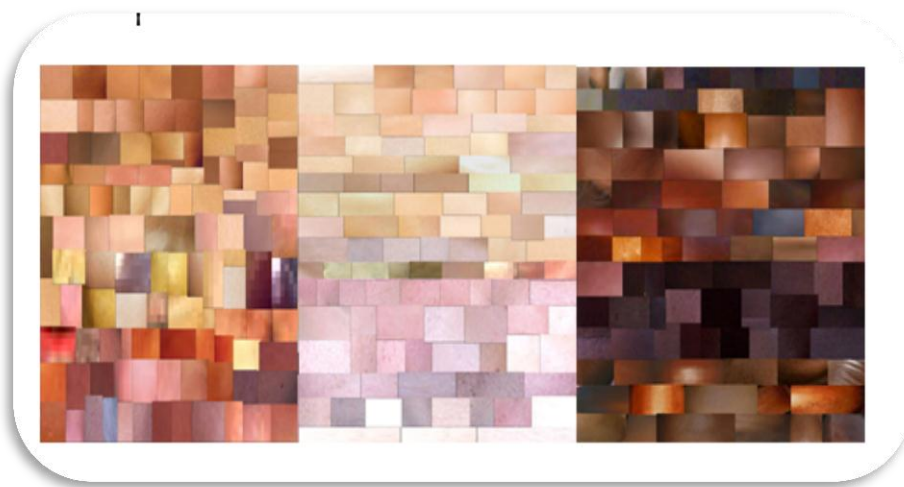


Figure 4.3 : base d'échantillons peau.

b) Création du model

Pour la création du model on suit les étapes suivantes :

1. Chargement des images de la base peau (*Figure 4.3*).
2. Transformation de l'espace RGB vers l'espace YCbCr.
3. Extraction les matrices Cb et Cr de chaque images et les transformés en vecteurs.
4. Stockage les vecteurs Cb et Cr dans une matrice à deux colonnes

$$\text{Mat} = \begin{pmatrix} \text{Cb} & \text{Cr} \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \end{pmatrix}$$

5. Calcul de la moyenne : $\text{moy} = \text{la_moyenne}(\text{mat}) = [M_{Cb} \quad M_{Cr}]$.

6. Calcul de la matrice de covariance :

$$\text{mat_cov} = E \{ [Cb - M_{Cb} \quad Cr - M_{Cr}] [Cb - M_{Cb} \quad Cr - M_{Cr}]^T \}$$

7. En utilisant les valeurs de la moyenne et la covariance, le model de couleur de peau peut être adapté à un modèle gaussien par [08] :

$$Y = \exp(-0.5 * [Cb - M_{Cb} \quad Cr - M_{Cr}] * \text{mat_cov}^{-1} * [Cb - M_{Cb} \quad Cr - M_{Cr}]^T)$$

La figure suivante illustre les modèles gaussiens de peau dans l'espace YCbCr :

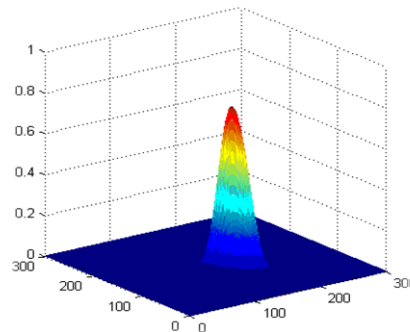


Figure 4.4 : Model gaussien de peau dans l'espace YCbCr.

c) Application du model

Après avoir appliqué le model gaussien sur l'image on obtient alors comme résultat une image probabiliste ou la valeur de chaque pixel désigne la probabilité pour que ce pixel soit une peau. À partir de cette image on peut obtenir une image binaire on utilisant un seuillage adaptatif où les « 1 » représentent les pixels de couleur peau et les « 0 » pour le reste des pixels.

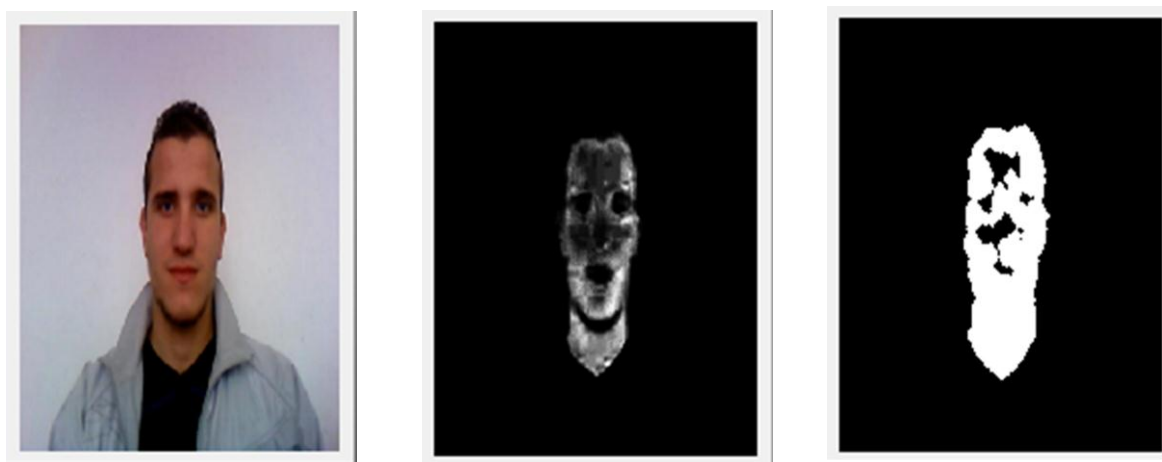


Image originale

Image probabiliste

Image binaire de peau

Figure 4.5 : Application du modèle gaussien simple

II.3.2 Modèle basé sur mélange gaussien

Pour pouvoir créer ce modèle il faut d'abord créer séparément trois bases d'échantillons de peau (noire, brune, blanche). Cette étape est suivie par le calcul de la gaussienne (Voir section § II.3.1) pour chaque base ce qui nous mène à un pixel qui a trois probabilités :

$$P(x) = 0.2 * \text{peau_noir} + 0.6 * \text{peau_brune} + 0.2 * \text{peau_blanc}$$



Image originale

Image probabiliste

Image binaire de peau

Figure 4.6 : Application du modèle mélange gaussien

II.4 Segment de peau :

Pour qu'on puisse traiter chaque segment de l'image binaire séparément on doit suivre le processus de traitement illustré par la *Figure 4.7* :

Amélioration des performances d'un système de détections de visages sur flux vidéo

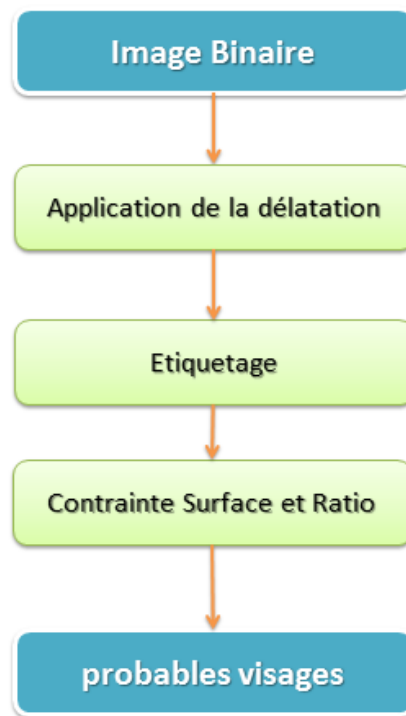


Figure 4.7: prétraitement des segments de peau.

II.4.1 Application de la dilatation:

Parfois l'image binaire obtenue dans la phase précédente porte des discontinuités et des trous ...etc, qui influe par la suite sur les résultats de l'étiquetage, les opérateurs morphologiques tels que la dilatation permettent de réduire ces problèmes.



Image binaire

Image dilatée

Figure 4.8: Application de la dilatation

II.4.2 Etiquetage :

Un autre algorithme aussi simple, mais fonctionnant de manière locale est l'algorithme à passes d'Haralick [25].

La première passe affecte des labels temporaires aux pixels, elle est définie par :

$$g(x, y) = \begin{cases} F_B & \text{si } b(x, y) = F_B \\ m, m = m + 1 & \text{si } \forall \{i, j \in M_S\} g(x - i, y - j) = F_B \\ g_{min}(x, y) & \text{sinon} \end{cases}$$

- Ou b est l'image d'entrée binarisée .
- $g(x, y) \in \{F_B, F_O\}$ est la valeur du pixel à la position (x, y) dans l'image binaire.
- F_B la valeur d'un pixel de fond (blanc).
- F_O la valeur d'un pixel de texte (noir).
- $g(x, y)$ contient l'étiquette du pixel à la position (x, y).
- m est l'étiquette courante affecté au prochain nouveau label.
- M_S est le masque des voisins (celui de parcours avant dans ce cas).

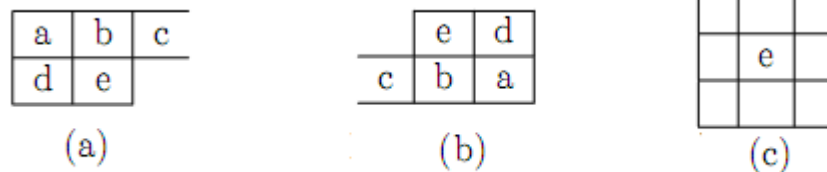


Figure 4.9 : Masques de voisinage de e pour : (a) le parcours avant ; (b) le parcours arrière ;(c)voisin de e

Les masques de voisinage définissent les pixels qui ont déjà été analysés lors du parcours courant, dans le cas d'un parcours avant, le masque est celui de la **figure 4.9 (a)**, dans le cas d'un parcours arrière celui de la **figure 4.9 (b)**.

La première passe va donner des étiquettes temporaires à g de telle sorte que chaque pixel possède la plus petite étiquette de ses voisins qui ont déjà été analysés ou une nouvelle étiquette s'il ne possède aucun voisin.

Une fois les étiquettes temporaires affectées à g, différentes passes sont effectuées sur g jusqu'à ce que plus rien ne soit modifié :

$$g(x, y) = \begin{cases} F_B & \text{si } g(x, y) = F_B \\ \min[\{g(x - i, y - j) | i, j \in M\}] & \text{sinon} \end{cases}$$

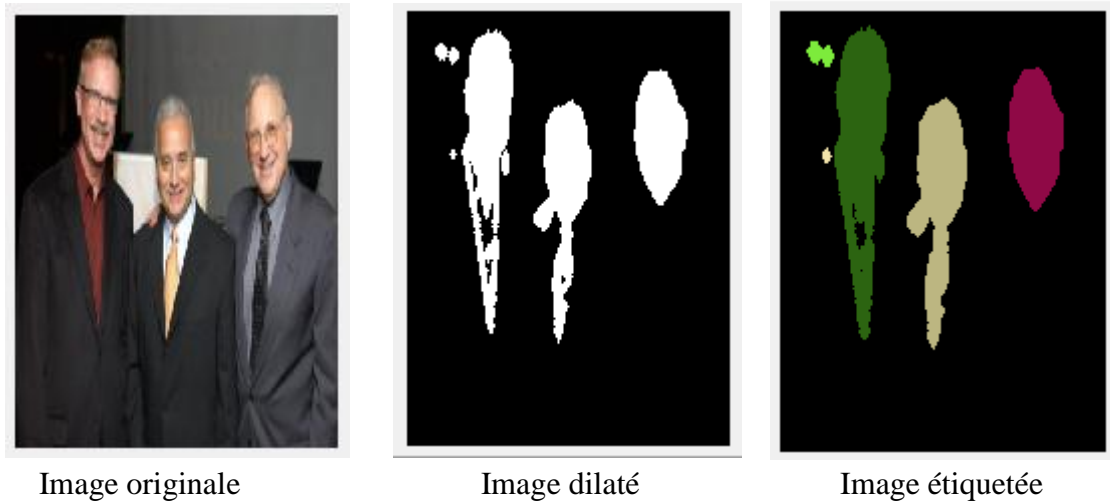


Figure 4.10: Résultats de l'étiquetage

II.4.3 Contraintes de surface et ratio

II.4.3.1 Contrainte de surface

La surface d'un segment est calculée par nombre de pixel peau appartenant à cette région. La hauteur et la largeur du segment sont calculées à partir des coordonnées du point supérieur gauche (X_1, Y_1) et inférieur droit (X_2, Y_2) du segment :

$$\text{Surface} = \sum(x, y) = 1; \quad \{x \in (X_1, X_2), y \in (Y_1, Y_2)\}.$$

La contrainte surface est introduite pour éliminer les segments de régions négligeables (trop petits). Tout segment présentant une surface plus petite que le seuil fixé est éliminé.

$$\text{Seuil Surface} \geq 150$$

II.4.3.2 Le Ratio

C'est le rapport entre la largeur et la hauteur du segment. L'intervalle convenable utilisé pour décider par rapport à la contrainte du Ratio est :

$$0.8 < \text{Ratio} < 2 \quad [26]$$



Image avant Application des contraintes Image après Application des contraintes

Figure 4.11 : Application des contraintes de surface et de ratio.

II.4.4 Probable visage

Après avoir appliqué les contraintes surface et ratio il nous reste que les segments peau qui peuvent représenter un visage et qui passant par le réseau de neurone pour une décision visage / non visage.

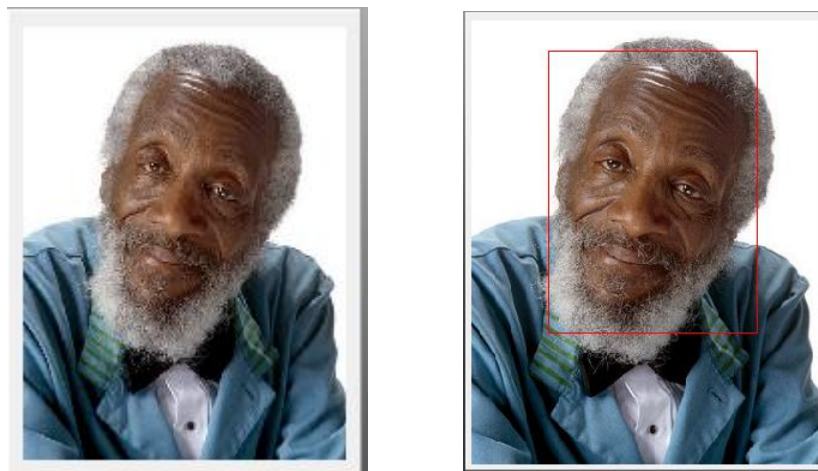


Image originale

probable visage

Figure 4.12 : extraction des visages probables.

II.5 Application du réseau de neurones

Comme il est déjà mentionné l'application des RNA consiste à décider visage ou non visage pour chaque segment en entrée, en effet la décision de ce dernier est exprimé par le taux d'appartenance de ce segment à une des classes de la base d'apprentissage (visage/non visage).

La **Figure 4.13** illustre le processus de détection de visages par réseau de neurones

Amélioration des performances d'un système de détections de visages sur flux vidéo

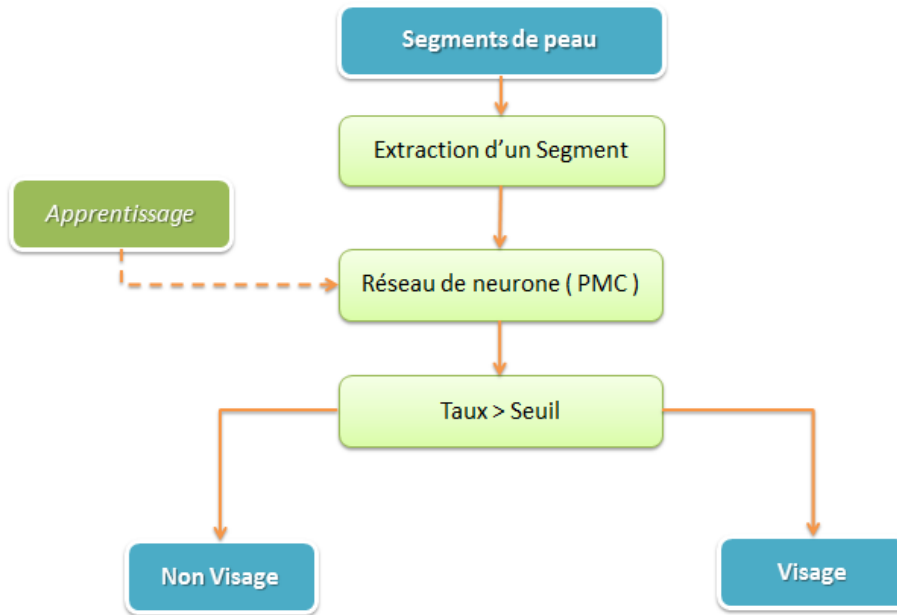


Figure 4.13 : le processus de la décision visage /non visage par PMC

Dans cette phase de développement on à utiliser le PMC comme un outil de décision cet outil est hiérarchisé comme suit :

Couche d'entrée : 600 neurones.

Couche cachée : 100 neurones.

Couche sortie : 1 neurone (visage/non visage)

II.5.1 L'apprentissage du réseau PMC

L'apprentissage est une étape nécessaire pour faire entrainer le réseau, pour cela on à utiliser une base de donnée non visage de 403 images et une base de donnée visage international COTADAS [27] de 344 images. Chaque image de ces bases sera redimensionnée en taille de (30*20). Les résultats de la phase d'apprentissage sont illustrés par les tableaux 4.1 et 4.2

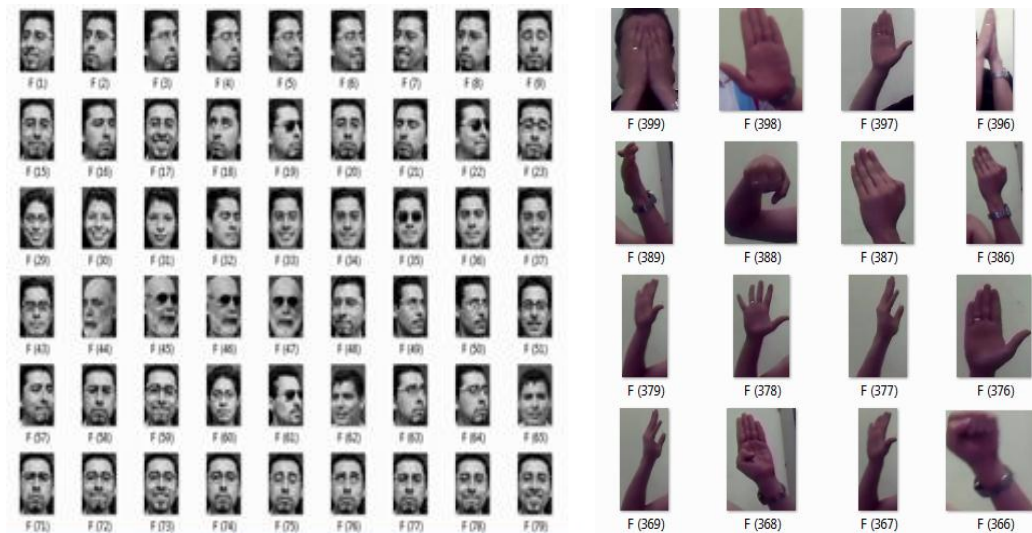


Figure 4.14 : Quelque échantillon de la base d'apprentissage

| Nombre d'itérations | Nombre de neurone de la couche caché | Erreur maximale Obtenu | Taux d'apprentissage de la BDD |
|---------------------|--------------------------------------|------------------------|--------------------------------|
| 500 | 30 | 0.0271 | 43.98 % |
| 500 | 50 | 0.0223 | 49.56 % |
| 500 | 100 | 0.0151 | 65.61% |

Tableau 4.1 : Taux d'apprentissage pour 500 itérations.

| Nombre d'itérations | Nombre de neurone de la couche caché | Erreur maximale Obtenu | Taux d'apprentissage de la BDD |
|---------------------|--------------------------------------|------------------------|--------------------------------|
| 5000 | 30 | 0.0091 | 59.55% |
| 5000 | 50 | 0.0082 | 67.82% |
| 5000 | 100 | 0.0050 | 85.35% |

Tableau 4.2 : Taux d'apprentissage pour 5000 itérations.

II.5.2 Test du réseau PMC :

C'est l'étape de la décision dont le réseau va décider de l'appartenance d'une section peau à la class visage ou non selon la sortie obtenu. La fonction de transfert utilisée (sigmoïde)

nous permet d'obtenir une valeur comprise entre $[0 \ 1]$, un seuil fixé à 0.5 nous permet de juger un segment comme étant un visage ou non.

La figure ci-dessous montre l'application du réseau de neurone sur une image couleur

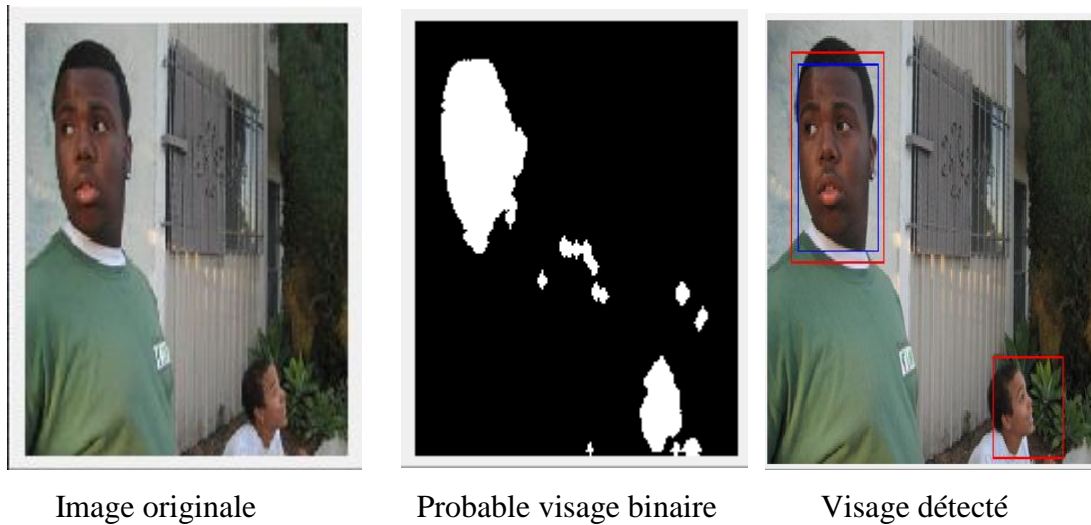


Figure 4.15 : Application du réseau de neurone.

III. Implémentation :

On va présenter dans cette partie l'implémentation de notre système «*DFACE*», et les différents composants nécessaires à son **fonctionnement**, l'environnement de développement **Matériel** (caractéristiques techniques de l'ordinateur et du système d'acquisition utilisé) et **logiciel** (système d'exploitation, langages de programmation...etc.) et les bases de données, suivi d'une illustration des résultats obtenus et leurs interprétations.

III.1 Environnement de développement

III.1.1 Environnement de développement matériel

a) Pc utilisé au cours du développement

- ⊕ Processeur : Intel Pentium (R) Dual-core (2.0 GHZ).
- ⊕ RAM : 3 GB.
- ⊕ Disque dur : 250 Go.

b) Caractéristiques du matériel d'acquisition

2. CyberLink YouCam

- ✓ Résolution maximale : 307200 pixels.
- ✓ Résolution minimale : 19200 pixels.
- ✓ Interface : intégrer.

III.1.2 Environnement de développement logiciel

On a utilisé pour la réalisation de notre projet la version 6 de l'environnement de développement C++builder, SDL component suite, VideoLab et Photoshop CS3.

a) C++ Builder

C++ Builder est le nouvel environnement de développement basé sur C++ In prise du Fort succès de Delphi, Borland a repris la philosophie, l'interface et la bibliothèque de composants visuels de ce dernier pour l'adapter depuis le langage Pascal Orienté Objet vers C++ répondant ainsi à une large faction de programmeurs peu enclins à l'utilisation du Pascal qu'ils jugent quelque peu dépassé [28].

Borland C++Builder est un environnement de programmation visuelle orienté objet permettant le développement des applications console Win32 (Windows 32bits) ou des programmes Win32GUI (Graphic Interface User) .En utilisant C++Builder, on peut créer de puissantes applications Windows.

C++Builder propose une bibliothèque de classes complète appelée VCL (bibliothèque de composants visuels) et une suite d'outils de conception rapide d'applications (RAD), Sous Builder Le programme n'est pas exécuté de façon séquentielle comme dans un environnement classique. Il s'agit de programmation « événementielle », des séquences de programme sont exécutées, suite à des actions de l'utilisateur (clique, touche enfoncée,.. etc.) détectées par WINDOWS.

Voici quelques-unes des caractéristiques de C++ builder :

- ✓ Programmation orienté objet.
- ✓ Outils visuels bidirectionnels.
- ✓ Possibilité d'utilisation des bibliothèques externe.
- ✓ Compilateur produisant des codes natifs.
- ✓ Traitement complet des exceptions.
- ✓ Possibilité de créer des exécutables.
- ✓ Bibliothèque de composants extensibles.
- ✓ Débogueur graphique intégré.

Et en plus le c++builder regroupe toute la puissance de c++.

La figure 4.16 permet une vue globale sur l'interface de la version 6 de l'environnement de développement c++builder.

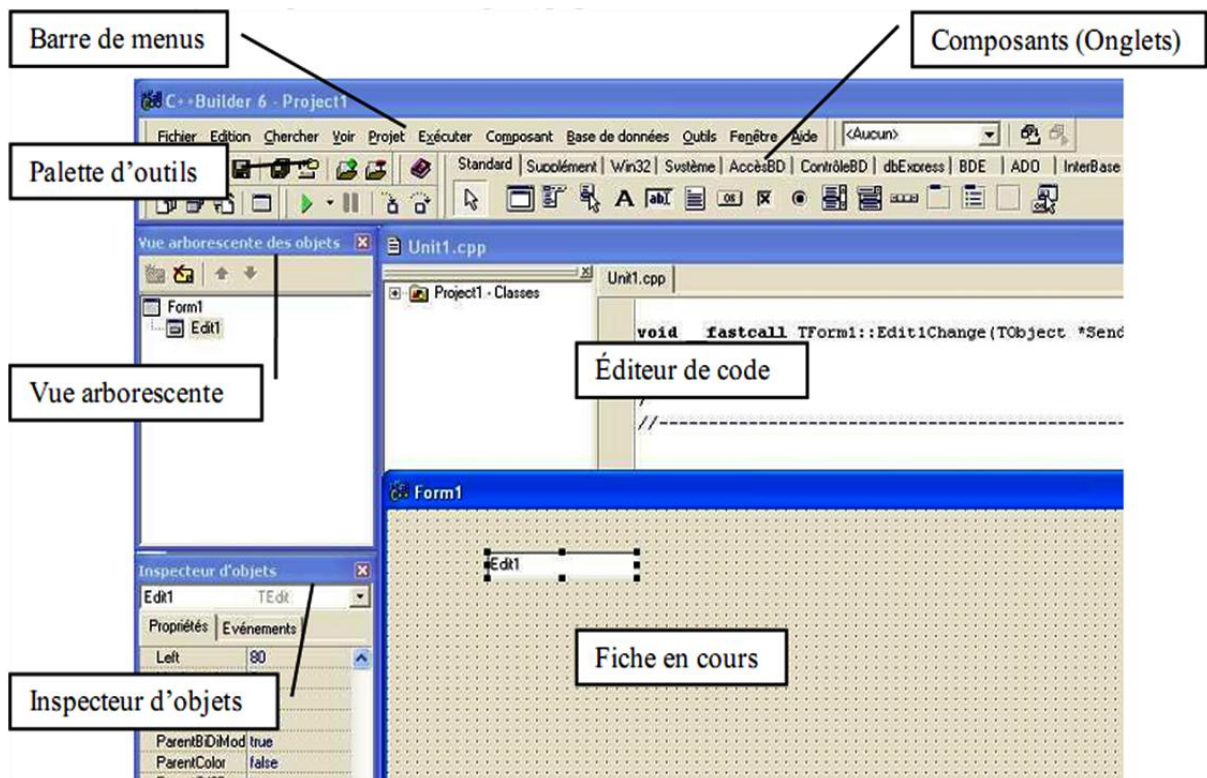


Figure 4.16 : L'interface de C++builder.

b) SDL component suite

le SDL component suite est une collection des composants scientifiques et informatique, la suite entiere contient à peu près 50 unités couvrant une large gamme de besoin en science et en génie la suite et disponible est en Delphi et en c++ builder.

Le SDL component suite est constitué de plusieurs packages qui peuvent être utiliser indépendamment, les packages utiliser dans notre application sont : mathpack, guipack, basepack.[29]

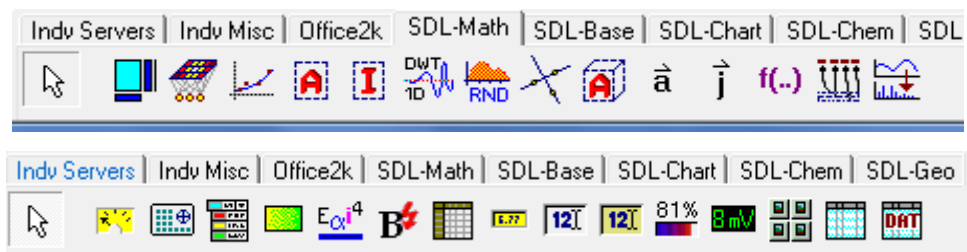


Figure4.17 : le composants SDL

c) vidéo lab

C'est un composants de Delphi / C + + Builder de la bibliothèque VCL pour le traitement vidéo. Permet des manipulations visuelles complexes et rapides avec un zéro de lignes de programmation. La bibliothèque prend en charge un très grand nombre de formats et de technique. Il comprend des filtres, transformations, la capture et la lecture des composants bûcherons, des mélangeurs, ...etc. Gratuit pour un usager non-commercial. [30]



Figure4.18: Composants videolab.

IV. Base de données image [31]

Pour l'évaluation de notre système on a utilisés comme base de teste des echantillons des bases d'images internationales BAO ,CALTECH et VAILD.

BAO : constituée de 140 images d'un seul visage et 221 images de plusieurs visages.

CALTECH : constituée de 450 images complexes.

VAILD: base audio visuel.



Figure 4.19 : Quelques échantillons de la base « BAO Faces ».



Figure 4.20: Quelques échantillons de la base « Caltech Faces ».

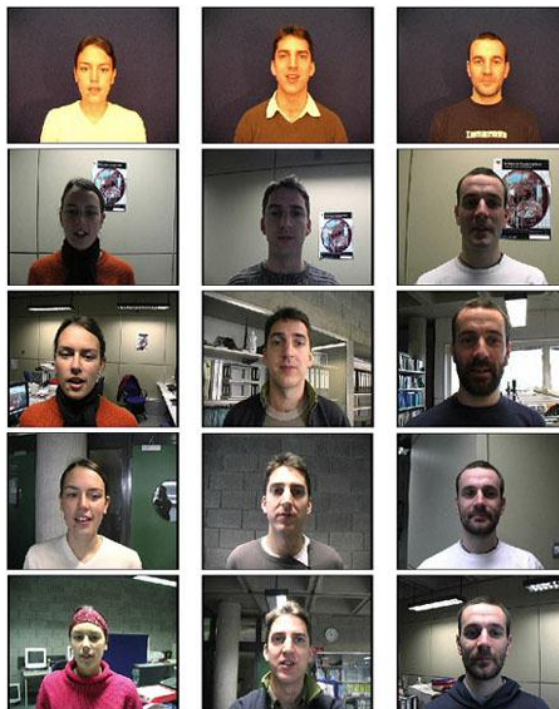


Figure 4.21 : Quelques échantillons de la base « Valid Faces ».

V. Présentation de l'application

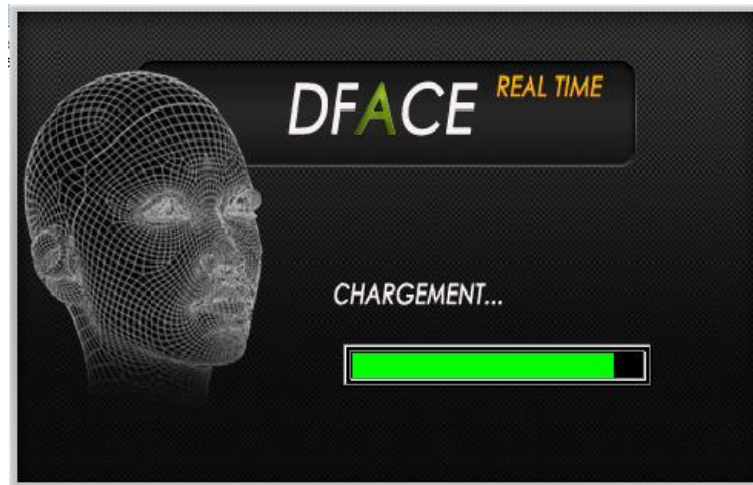


Figure 4.22: Fenêtre de lancement de « DFACE ».



Figure 4.23: Fenêtre principale de « DFACE ».

Comme le montre la figure 4.23, la fenêtre principale conduit à trois types de traitement :

Amélioration des performances d'un système de détections de visages sur flux vidéo

- 1 : Ce bouton nous permet d'accéder à un environnement (interface) consacré pour le traitement sur les images fixes
- 2 : Interface de traitement de vidéo chargé à partir du disque dur.
- 3 : Détection des visages en direct à partir d'une webcam.
- 4 : À propos de l'application.



Figure 4.24 : Fenêtre de détection de visage sur image fixe.

- 1 : Charger une image à partir d'un répertoire
- 2 : Détection de peau par model gaussien simple.
- 3 : Détection de peau par model mélange gaussien.
- 4 : Apprentissage du réseau de neurone

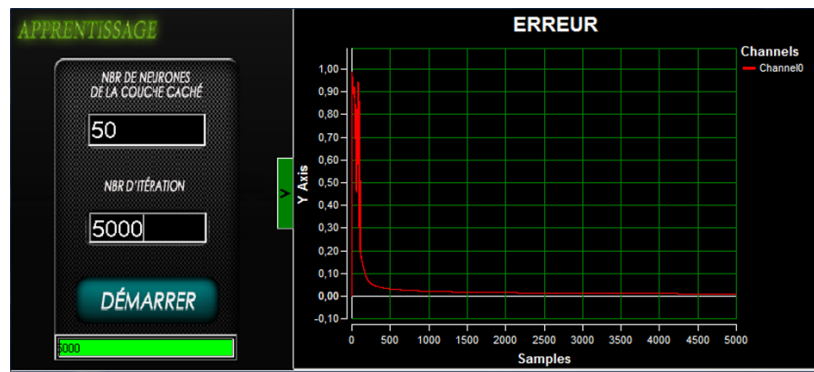


Figure 4.25 : Fenêtre d'apprentissage du réseau de neurones

- 5** : Retour au menu principal
- 6** : Affichage de l'image chargée.
- 7** : Affichage de l'image probabiliste de peau.
- 8** : Affichage de l'image binaire de peau.
- 9** : Affichage de l'image dilaté.
- 10** : Affichage de l'image étiquetée.
- 11** : Affichage de l'image contenant les probables visages.
- 12** : Affichage de l'image contenant les visages détectés.

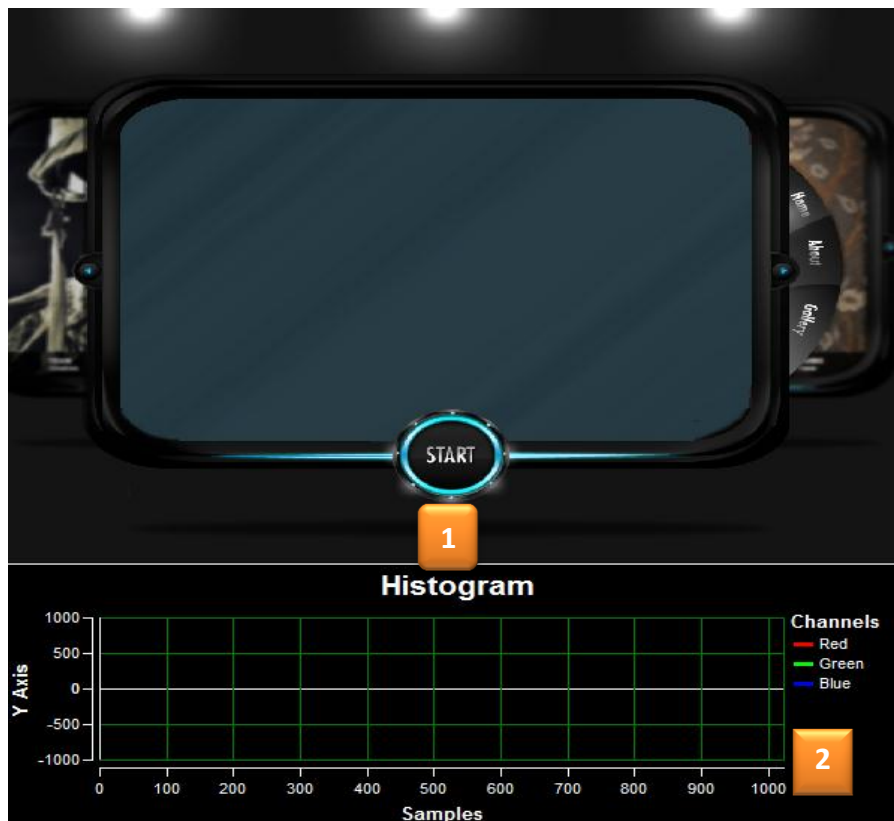
— Rectangle rouge : Probable visage.

— Rectangle bleu : visage détecté.



Figure 4.26 : Fenêtre de détection de visage sur un flux vidéo archivé.

- 1** : Charger une vidéo à partir d'un répertoire.
- 2** : Démarrer la détection des visages sur le flux vidéo.
- 3** : Quitter la fenêtre.



Amélioration des performances d'un système de détections de visages sur flux vidéo

Figure 4.27 : Fenêtre de détection de visages sur flux vidéo en direct.

- 1** : Lancement et arrêt de la capture vidéo.
- 2** : Affichage de l'histogramme de chaque image de la vidéo.



Figure 4.28: Fenêtre à propos.

VI. Tests et résultats

Pour évaluer les performances de notre système on a besoins de le tester sur deux bases images et vidéo qui porte une variation de caractéristiques (nbre de segments de peau, nbre de visages, la complexité de la scène...etc).

Les performances sont exprimées par deux valeurs :

$$\text{Taux de détections} = \frac{\text{Nbr de visages détectés}}{\text{Nbr de visages réels}}$$

$$\text{Taux de fausses détections} = \frac{\text{Nbr de fausses détections}}{\text{Nbr de détections}}$$

VI.1 Tests sur images fixes

Nous commençons par évaluer les performances des systèmes « DFACE » sur les images fixes issues des bases précédemment citées ou enregistrer par nos soins.

Amélioration des performances d'un système de détections de visages sur flux vidéo

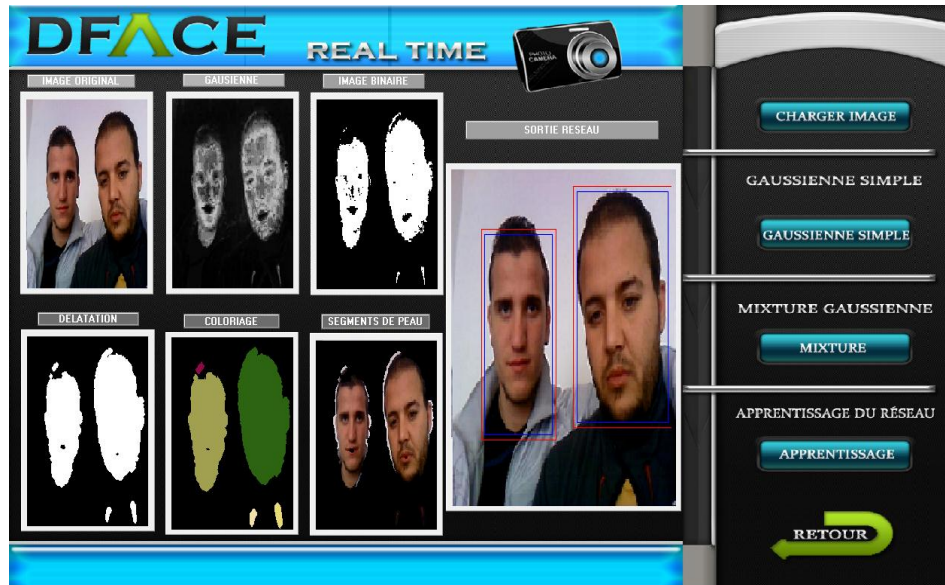


Figure 4.29: Test 01[acquise par webcam].

| Nbr Visages réels | Nbr Détections | Nbr Visages détectés | Nbr Visages non détectés | Nbr Fausse détection | Taux détection | Taux fausse détection |
|-------------------|----------------|----------------------|--------------------------|----------------------|----------------|-----------------------|
| 2 | 2 | 2 | 0 | 0 | 100% | 0% |

Tableau 4.3: Résultats du Test 01.

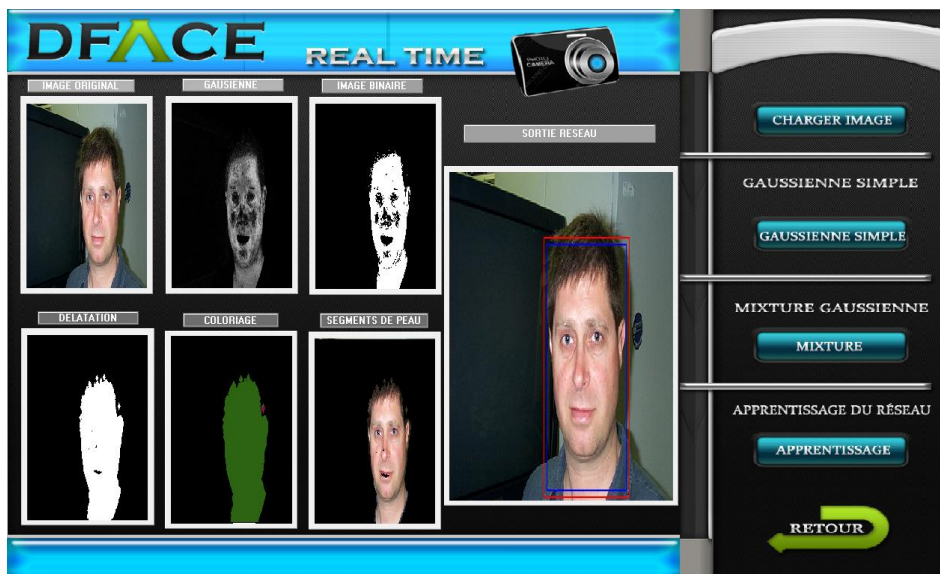


Figure 4.30: Test 02[base caltech].

| Nbr Visages réels | Nbr Détections | Nbr Visages détectés | Nbr Visages non détectés | Nbr Fausse détection | Taux détection | Taux fausse détection |
|-------------------|----------------|----------------------|--------------------------|----------------------|----------------|-----------------------|
| 1 | 1 | 1 | 0 | 0 | 100% | 0% |

Amélioration des performances d'un système de détections de visages sur flux vidéo

Tableau 4.4 : Résultats du Test 02.

Lors de l'analyse des résultats obtenue on peut classifier les erreurs commise par le système en 4 classes: fausses détections due au réseau de neurones, fausses détections due à la segmentation de peau, fausses détections due à la dilatation.

VI.1.1 Fausses détections due au réseau de neurones

Dans cette classe on peut trouver deux types d'erreurs :

- Erreur de détection de visages : c'est la détection de segment comme étant un visage alors qu'il n'est pas. (Figure 4.31).
- Non détection de visage : dans le cas où un segment correspond à un visage, mais ce dernier n'est pas détecté par le système. (Figure 4.32).



Figure 4.31: Test 03[base valid].

| Nbr Visages réels | Nbr Détections | Nbr Visages détectés | Nbr Visages non détectés | Nbr Fausse détection | Taux détection | Taux fausse détection |
|-------------------|----------------|----------------------|--------------------------|----------------------|----------------|-----------------------|
| 1 | 2 | 1 | 0 | 1 | 100% | 50% |

Tableau 4.5 : Résultats du Test 03.



Figure 4.32: Test 04[base BAO].

| Nbr Visages réels | Nbr Detections | Nbr Visages détectés | Nbr Visages non détectés | Nbr Fausse détection | Taux détection | Taux fausse détection |
|-------------------|----------------|----------------------|--------------------------|----------------------|----------------|-----------------------|
| 4 | 4 | 3 | 1 | 0 | 75% | 0% |

Tableau 4.6 : Résultats du Test 04.

VI.1.2 Fausses détections due à la segmentation de peau

Considéré dans le cas où l'image contient des objets avec des couleurs proches à celles de la peau (bois ou peinture mural...etc.), ces objets peuvent causer des problèmes de détections. (Figure 4.33).

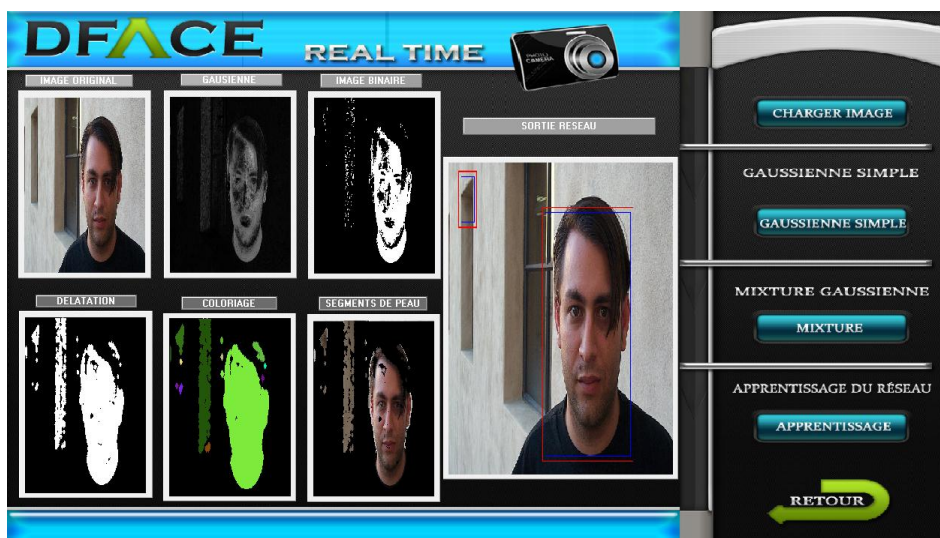


Figure 4.33: Test 05[base caltech].

| Nbr Visages réels | Nbr Détections | Nbr Visages détectés | Nbr Visages non détectés | Nbr Fausse détection | Taux détection | Taux fausse détection |
|-------------------|----------------|----------------------|--------------------------|----------------------|----------------|-----------------------|
| 1 | 2 | 1 | 0 | 1 | 100% | 50% |

Tableau 4.7 : Résultats du Test 05.

VI.1.3 Fausses détections due à la dilatation

L'opérateur morphologique de dilatation peut générer des problèmes lors de la détection de visage tel que l'intersection de deux segments peau adjacents. (Figure 4.34).

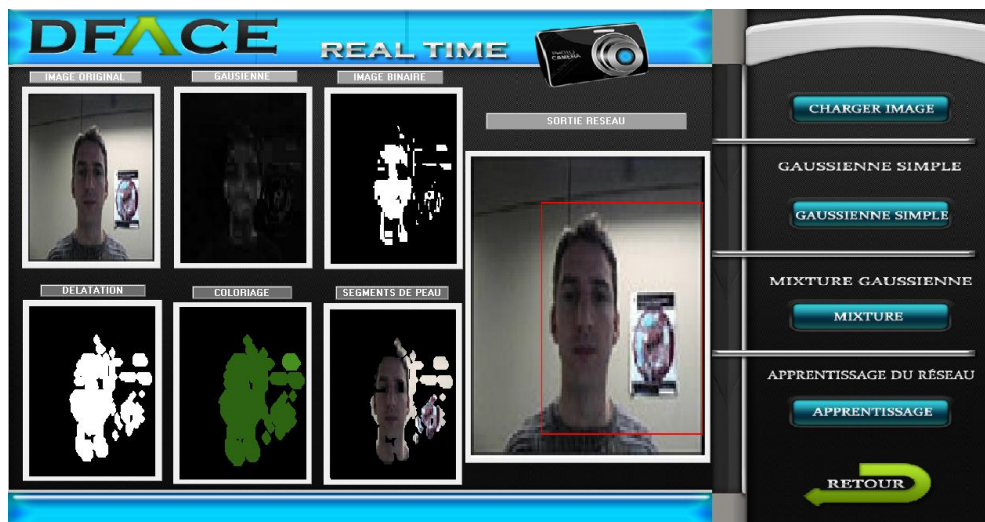


Figure 4.34: Test 06[base valid].

| Nbr Visages réels | Nbr Détections | Nbr Visages détectés | Nbr Visages non détectés | Nbr Fausse détection | Taux détection | Taux fausse détection |
|-------------------|----------------|----------------------|--------------------------|----------------------|----------------|-----------------------|
| 1 | 0 | 0 | 0 | 0 | 0% | 0% |

Tableau 4.8 : Résultats du Test 06.

VI.2 Test sur vidéo archivé

Les vidéos utilisées dans notre système sont caractérisés par :

- dimension de 152*200 .
- Notre système peut traiter un nombre de 15 images /seconde, ce nombre est en inversement proportionnel avec la dimension de la vidéo et le nombre de segment peau à traiter (si la dimension ou le nombre de segment augment, le nombre d'image traiter par second diminue).

Amélioration des performances d'un système de détections de visages sur flux vidéo

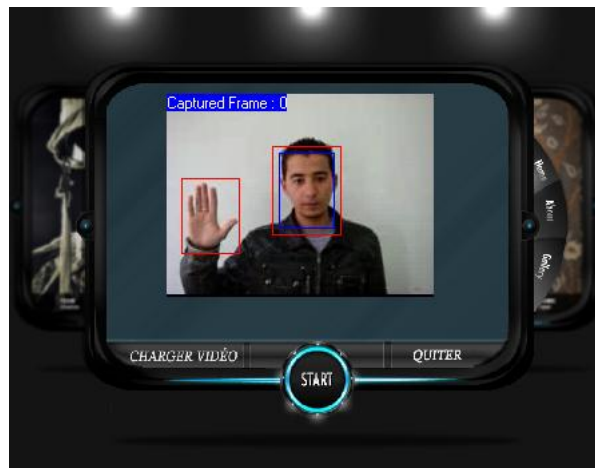


Figure 4.35: Test 07.

| Nbr d'images dans la vidéo | Nbr Visages réels dans la vidéo | Taux détection | Taux fausse détection |
|----------------------------|---------------------------------|----------------|-----------------------|
| 333 | 1 | 100% | 0.0015% |

Tableau 4.9 : Résultats du Test 07.

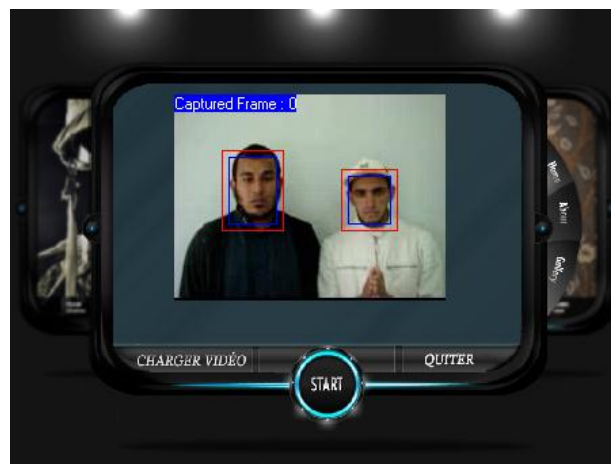


Figure 4.36: Test 08.

| Nbr d'images dans la vidéo | Nbr Visages réels dans la vidéo | Taux détection | Taux fausse détection |
|----------------------------|---------------------------------|----------------|-----------------------|
| 225 | 2 | 70.84% | 0% |

Tableau 4.10 : Résultats du Test 08.

VI.2.1 Comparaison entre DFACE et PFE 2010 [32]

Le tableau ci-dessous illustre une comparaison entre les résultats obtenus par les deux systèmes.

| Vidéo | Logiciel | Num de frame | Nbr de détection | Nbr de visage détecté | Nbr de fausse détection | Taux de détection | Taux de fausse détection | Temps d'exécution Images/sec |
|-----------|-------------|-----------------|------------------|-----------------------|-------------------------|-------------------|--------------------------|------------------------------|
| Vidéo (1) | Projet 2010 | 1 à 12 | 1 | 1 | 0 | 100 % | 0 % | 0.63 |
| | DFACE | | 1 | 1 | 0 | 100 % | 0 % | 0.066 |
| Vidéo (2) | Projet 2010 | 1,2,3,4, 5 et 7 | 2 | 2 | 0 | 100 % | 0 % | 0.61 |
| | DFACE | | 2 | 2 | 0 | 100 % | 0 % | 0.066 |
| | Projet 2010 | 6,8 | 1 | 1 | 0 | 50% | 0% | 0.61 |
| | DFACE | 6.8 | 2 | 2 | 0 | 100% | 0% | 0.066 |

Tableau 4.11: Résultats de comparaison entre DFACE & PFE 2010

VI.3 Test sur vidéo en direct

En plus de la détection du visage sur vidéo archivée le système « *DFace* » nous offre une détection de visage en temps réel à partir d'une webcam ou une caméra. Les résultats obtenus sont très prometteur comparé aux systèmes existants en général et au système conçu dans le cadre de PFE 2010 en particulier.

VII. Conclusion

Après avoir conçu le système DFACE et touché ses points fort et ses points faible, on peut conclure que pour augmenter la fiabilité de ce dernier il faut améliorer la qualité d'apprentissage du réseau PMC en mettant en œuvre une base d'apprentissage plus riche.

Conclusion Générale

Dans ce travail, il était question de réaliser un système de détection et de visages basé sur le réseau de neurones perceptrons multicouches et la segmentation de régions de couleur de la peau.

Le fond de ce travail se concentre sur la détection de visages dans une vidéo, ou acquise directement via une webcam (live) par hybridation de deux méthode : la mixture gaussienne et les réseaux neuronaux, La première phase consiste à détecter les zones contenant la couleur chair humaine par mixture de modèles gaussiens l'objectif principal de cette étape est la limitation de l'espace de recherche de l'objet visage. La seconde étape consiste à décider de l'appartenance de la zone peau à la classe des visages ou des non visages par le biais des réseaux de neurones.

Après avoir implémenté le module nous avons validé et analysé ses performances sur des bases d'images fixes et vidéo où nous avons aboutis à des résultats intéressants compte tenu du taux de détection. En ce qui concerne le temps de traitement, la vérification d'une image ne dépasse pas en moyenne les 0.1s pour une vidéo de résolution 150x200.

Les résultats obtenus sont très promoteur (traitement de 15 images par second) et répandent aux exigences de détection de plusieurs applications de vidéo surveillance à basse résolution.

Pour aboutir à une cadence de traitement plus élevée tel que 25 ou 30 images par seconde pour un system de détection de visages avec une résolution importante il serait intéressant d'utiliser l'approche multithread pour traiter chaque probable visage indépendamment ainsi que l'utilisation des méthodes de suivi pour minimiser l'espace de localisation et aussi intégrer d'autre approche pour éliminer les erreurs présentées dans le dernier chapitre et augmenter les performances du système en améliorant la qualité d'apprentissage .

Bibliographie

- [1] D.Khebab, Belkhefha.M « Thème Système de détection du visage humain basé sur la méthode SVM », Mémoire de fin d'études, 2007.
- [2] R .OUAFEK, M.MEGHALSI « Thème La détection des visages dans une image », Mémoire de fin d'études, 2009.
- [3] O.Bernier « vision par ordinateur pour les interfaces visage et gestes », Mémoire pour l'obtention d'une Habilitation à diriger des recherches, université de pierre et marie curie, 2009.
- [4] T.Phuong, T. Hoang Lan « Amélioration des performances d'un système de détection de visage », exposé, Institut de la francophonie pour l'informatique (IFI) , 2006.
- [5] http://www.evitech.com/files/Evitech_AViRS08_04.pdf, avril 2011.
- [6] M.Clergue « Réseaux de Neurones Artificiels » cours.
- [7] L.Liang, A.Haizhou et GuangyouXu « Face Detection based on Template Matching and Neural Network Verification», 1999.
- [8] E. Osuna, R. Freund, et F. Girosi, « Training Support Vector Machine An application to Face Detection », IEEE Proc, Int. Conf. Computer Vision and Pattern Recognition, 6, 1997.
- [9] F. Samaria: "Face segmentation for identification using Hidden Markov Models", in British machine vision conference, 1993.
- [10] <http://www.techno-science.net/?onglet=glossaire&definition=7376>, Janv 2011.
- [11] www.europaschool.net/static/formation/pdf/formats_son_video.pdf, Fév 2011.
- [12] <http://www.commentcamarche.net/contents/video/format-mpeg.php3>, Fév 2011.
- [13] S. Thon et E. Remy (IUT de Provence) « Imagerie Numérique Représentation et codage des images », Cours, 2003.
- [14] http://en.wikipedia.org/wiki/CIE_1931_color_space, Mai 2011.
- [15] D. Désiré SIDIBE, « Une technique de relaxation pour la mise en correspondance d'images », Thèse pour obtenir le grade de DOCTEUR DE L'UNIVERSITÉ MONTPELLIER II, le 07 décembre 2007.
- [16] <http://fr.wikipedia.org/wiki/YUV>, MAI 2011.

- [17] <http://www-igm.univ-mlv.fr/~dr/XPOSE2002/Neurones/index.php>, MAI 2011.
- [18] <http://www.peoi.org/Courses/Coursesfr/neural/EL150FR.html>, MAI 2011.
- [19] <http://www.peoi.org/Courses/Coursesfr/neural/neural3.html>, MAI 2011.
- [20] <http://www.sylbarth.com/nn.php>, MAI 2011.
- [21] <http://cict.fr/~stpierre/reseaux-neuronaux/node9.html>, JUIN 2011.
- [22] A.KHATIR « Thème Etude comparative des modèles des machines asynchrones utilisés endiagnostic des défauts », université de setif, Mémoire Pour l'obtention du Grade Magister En Electrotechnique, 2009 .
- [23] <http://www-inrev.univ-paris8.fr/extras/Michel-Bret/cours/bret/cours/conn/c3.htm>, JUIN 2011.
- [24] http://eliot.utc.fr/Docs/NOTICE_GENERALE/000/0000/00009304/IA01_cours13_neurones_artificielles_1.0.0.pdf, JUIN 2011.
- [25] R. M. Haralick, « Some neighborhood operations », pages 11–35. Plenum Press, 1981.
- [26] Merabti.M, Farjallah.R, « Système de détection de visage par méthode hybride », Mémoire d'ingénieur en informatique, Université de Guelma 2009.
- [27] <http://www.vision.caltech.edu/html-files/archive.html>, Avril 2011
- [28] http://eric.univ-lyon2.fr/~ricco/cours/cours/intro_cpp_builder.pdf, juin 2011.
- [29] http://www.lohninger.com/download_sdlsuite.html, avril 2011.
- [30] http://www.mitov.com/Video_Lab_Components/videolab.html.avril 2011.
- [31] <http://www.face-rec.org/databases>,juin 2011.
- [32] Khammari.M, H.Amroune, Système de Détection et Suivi e Visages sur Images Fixes et Dynamique par Mixture Gaussienne et Réseau Neuronal, PFE pour obtention le grade master 2,2010.