

---

**MÉMOIRE DE FIN D'ETUDES**

Pour l'obtention du diplôme

**MASTER EN INFORMATIQUE**

**Spécialité : Systèmes d'Informations**

Thème

---

**Identification faciale biométrique  
Par les Chaines de Markov Cachées  
(HMM)**

---

**Présenté par :**

- ZIANI khalida
- LAROUÏ H. karima

**Membres du jury :**

- |                            |            |
|----------------------------|------------|
| ❖ Mr. YAHIAOUI Yasser      | Président  |
| ❖ Mr. ABDELMOUMENE Halima  | Examineur  |
| ❖ Mr. BENAÏSSA Bedr-Eddine | Rapporteur |

**Année Universitaire : 2017/2018**

## Remerciement



**E**n tout premier lieu, nous remercions Allah le tout puissant, à la sagesse et au savoir infinis, " Gloire à Toi ! Nous n'avons de savoir que ce que Tu nous as appris. Certes c'est Toi l'Omniscient, le Sage."(Sourate al-Baqarah, verset 32).

Nous exprimons notre reconnaissance et nos plus vifs remerciements à notre encadreur **Mr .Benaissa Bedr Eddine** pour ses conseils, son aide, son soutien dans les moments difficiles. Nous le remercions pour nous avoir guidés pour l'élaboration de ce mémoire et pour le choix judicieux du thème.

Nos remerciements s'adressent également à tous les enseignants qu'ils ont nous formés durant nos études.

Enfin nos meilleurs et vifs remerciements s'adressent aux membres du jury pour avoir accepté d'évaluer ce projet.





## *Dédicace*

*Dieu me suffit, quel excellent protecteur*

A

*Ma mère & mon père*

*Vous êtes pour moi une source de vie car sans vos sacrifices, votre tendresse et votre affection je ne pourrais arriver jusqu'au bout. Je me réjouis de cet amour filial. Que Dieu vous garde afin que votre regard puisse suivre ma destinée.*

A

*Mon frère & Mes sœurs*

A

*A tous mes proches ;*

*Je dédie ce travail.*

*ZIANI KHALIDA*



## Dédicace

*Je rends grâce à dieu de m'avoir donné le courage et la volonté ainsi que la conscience d'avoir pu terminer mes études.*

*Je dédie ce modeste travail :*

*À mes très chers parents Mon père : BAHOUS & Ma mère : SETTI.*

*en reconnaissance de leurs amour, encouragement et affection qu'ils m'ont prodigués durant mes études. Que dieu me les garde.*

*À mes frères Mohamed et Abderrahmane.*

*À ma sœur Faiza.*

*À toute mes amies*

*À toute ma promotion chacun son nom.*

*À tous ceux et celles qui m'aiment et qui m'ont aidé de loin ou de près.*

*Laroui hadja Karima.*

## Résumé

Cette étude concerne l'application d'une approche markovienne pour identifier le visage d'un individu. L'objectif principale est de simplifier le déroulement de l'identification biométrique qui consiste à la recherche de l'identité requête parmi l'ensemble des individus préenregistrée. Après étude des différentes méthodes, on a opté pour une méthode markovienne utilisant l'algorithme Baum-Welch, pour la détection de caractéristique visage,. La validation de cette technique est réalisée à l'aide des banques d'images XM2VTS(Extended Multi Modal Verification for Teleservices and Security application), contenant les photos de 400 personnes. Les résultats obtenus d'après les tests montrent que la technique est motivante et nous ont encourage à appliquer ultérieurement l'identification en temps réel. Nous avons mis en évidence les différentes difficultés inhérentes à la reconnaissance automatique du visage, notamment la variation de pose et l'invariance à l'illumination.

Mots clefs : Identification, Approche Markovienne, Algorithme Baum-Welch

## Abstract

This study concerns the application of a Markovian approach to identify the face of an individual. The main objective is to simplify the course of the biometric identification which underlies the identity request among the whole of the individuals preregistered. After the study of various methods, this work has opted the Markovian method using the Baum-Welch algorithm for the detection of the face characteristic. The validation of this technique is carried out using the image data bases XM2VTS(Extended Multi Modal Verification for Teleservices and Security application), containing the photographs of 400 people. The results obtained according to the tests show that the technique is motivating which encourages us to apply later as a technique of identification in real time. We have highlighted the different difficulties inherent in the automatic recognition of faces, particularly variation of pose and the invariance to enlightenment.

Keywords: identify, Markovian approach , Baum-Welch algorithm

## ملخص

تتعلق هذه الدراسة بتطبيق نهج ماركوف لتحديد وجه الفرد. الهدف الرئيسي منها هو تبسيط عملية تحديد الهوية البيومترية ، والتي تتضمن البحث عن الهوية المطلوبة بين الأفراد المسجلون مسبقا. بعد دراسة مختلف الطرق ، استقرينا على طريقة ماركوف باستخدام خوارزمية Baum-Welch ، للكشف عن خاصية الوجه. يتم التحقق من صحة هذه التقنية باستخدام صور من قاعدة الصور XM2VTS(Extended Multi Modal Verification for Teleservices and Security application) التي تحتوي على صور ل 400 شخص. تُظهر النتائج التي تم الحصول عليها من الاختبارات أن هذه التقنية تحفزنا وتشجعنا على تطبيق تقنية التعرف في الوقت الحقيقي. لقد سلطنا الضوء على الصعوبات المختلفة الكامنة في التعرف التلقائي على الوجه ، بما في ذلك الاختلاف في الوضعيات و عدم تغيير الإضاءة.

كلمات المفتاحية: تحديد, نهج ماركوف, خوارزمية Baum-Welch

Résumé .....	i
Tables des matières.....	iii
Table des figures.....	v
Liste des tableaux.....	v
Introduction générale.....	1
<b>Chapitre 1 : Etat de l'art des techniques de reconnaissance de visages</b>	
I. Introduction.....	5
II. Les approches globales.....	5
II.1 Eigenfaces.....	5
II.2 Fisherfaces.....	6
II.3 L'utilisation de Transformée en cosinus discrète DCT.....	7
II.4 Machine à Vecteurs de Support (SVM).....	8
II.5 Les réseaux de neurones (RNA).....	8
III. Les approches locales.....	8
III.1 Eigen Objects (EO).....	9
III.2 Elastic Buch Graph Matching.....	9
III.3 Couleurs.....	10
III.4 L'appariement du gabarit (Template Matching ).....	11
III.5 Modèle de Markov caché.....	12
IV. Tableau comparatif des avantages et inconvénients des méthodes de reconnaissance faciale.....	16
V. Conclusion.....	16
<b>Chapitre 2 : Chaîne de Markov Cachée</b>	
I. Introduction.....	17
II. Modèle de Markov .....	17
II.1 Les concepts de chaîne de Markov .....	17
II.1.1 Processus stochastiques.....	17
II.1.2 Processus de Markov .....	18
II.1.3 Chaîne de Markov à temps discret .....	18
III. Chaînes de Markov Cachées .....	20
III.1 Définition .....	19
III.2 Théorie bayésienne .....	21
III.2.1 La loi de X a priori.....	21
III.2.2 Loi de (X, Y) .....	22
III.3 Prérequis Statistiques .....	22
III.3.1 Moyenne .....	22
III.3.2 Ecart-type .....	22
IV. Problèmes fondamentaux .....	29
IV.1 Premier problème: L'évaluation .....	23
IV.2 Deuxième problème : Quelle est la X la plus probable ?.....	25
IV.3 Troisième problème : Apprentissage .....	27
V. Conclusion.....	28

**Chapitre 3 : Contribution et Conception**

I. Introduction.....	29
II. Contribution .....	29
II.1 Le système d'apprentissage.....	29
II.2 Le système de reconnaissance.....	30
III. Conception en UML .....	31
III.1 Diagramme Uses case.....	31
III.1.1 Le système d'apprentissage.....	31
III.1.2 Le système de reconnaissance .....	32
III.2 Diagramme de classe .....	34
IV .Conclusion.....	35

**Chapitre 4 : Implémentation et Discussion**

I. Introduction.....	36
II. Aspect matériel .....	36
III. Outils de développement.....	36
IV. But du logiciel.....	37
V. Architecture structurelle du système de reconnaissance proposé .....	37
VI. Modélisation .....	38
VI.1 Paramètres CMC .....	38
VI.1.1 Procédure Forward .....	38
VI.1.2 Procédure Backward .....	38
VI.2 Algorithme .....	39
VI.2.1 Apprentissage .....	39
VI.2.2 Reconnaissance.....	40
VII. L'interface de l'application .....	40
VII.1 L'interface principale.....	40
VII.2 L'interface de la base des données .....	41
VII.3 L'interface de la base de caractéristiques .....	42
VIII. Déroulement de l'application.....	43
VIII.1 L'apprentissage.....	43
VIII.1.1 Chargement de photo .....	43
VIII.1.2 Prétraitement de photo .....	44
VIII.1.3 Extraction de caractéristiques.....	44
VIII.1.4 Mémorisation de caractéristiques.....	45
VIII.2 Reconnaissance.....	46
IX. Résultats et Expérimentations .....	48
IX.1 Expérience.....	48
IX.1 .1 Orientation et pose.....	49
IX.1 .2 Conditions d'illumination.....	49
IX.1 .3 Qualité de l'image.....	50
IX.1 .4 Expressions faciales.....	50
IX.1 .5 Présence caractéristiques particulières (une barbe).....	51
X. Conclusion.....	53
Conclusion générale et perspective .....	54
Bibliographie.....	55

## Table des figures

<b>Fig.1.</b> Etapes de la reconnaissance du visage.....	2
<b>Fig.2.</b> Reconnaissance faciale par Eigenfaces.....	6
<b>Fig.3.</b> Reconnaissance faciale par Fisherfaces.....	7
<b>Fig.4.</b> Reconnaissance faciale par réseaux de neurones.....	8
<b>Fig.5.</b> Reconnaissance faciale par Elastic Buch Graph Matching.....	10
<b>Fig.6.</b> Reconnaissance faciale par la couleur de peau.....	11
<b>Fig.7.</b> Reconnaissance faciale par Template Matching.....	11
<b>Fig.8.</b> HMM de gauche à droite.....	12
<b>Fig.9.</b> Structure d'un HMM 2D.....	12
<b>Fig.10</b> Topologie de modèle.....	14
<b>Fig.11</b> Diagramme de création de modèle d'asymétrie faciale.....	15
<b>Fig.12.</b> Exemple de modélisation de Chaîne de Markov.....	19
<b>Fig.13.</b> Chaîne de Markov Cachée d'ordre 1.....	21
<b>Fig.14.</b> Séquence d'opérations pour forward.....	23
<b>Fig.15.</b> Séquence d'opérations pour Backward.....	24
<b>Fig.16.</b> Principe de fonctionnement de l'algorithme Viterbi.....	26
<b>Fig.17.</b> Les étapes de l'apprentissage.....	30
<b>Fig.18.</b> Les étapes de reconnaissance .....	30
<b>Fig.19.</b> Diagramme de cas d'utilisation de système d'apprentissage.....	31
<b>Fig.20.</b> Diagramme de cas d'utilisation de système reconnaissance.....	33
<b>Fig.21.</b> Diagramme de Classe .....	34
<b>Fig.22.</b> Architecture structurelle de notre système.....	37
<b>Fig.23.</b> Interface principale de l'application.....	41
<b>Fig.24.</b> L'interface de la base des données.....	42
<b>Fig.25.</b> base de caractéristiques.....	42
<b>Fig.26.</b> Charger la photo à rechercher.....	43
<b>Fig.27.</b> Message fin chargement.....	43
<b>Fig.28.</b> Prétraitement de photo.....	44
<b>Fig.29.</b> Extraction de caractéristiques.....	45
<b>Fig.30.</b> Mémorisation de caractéristiques .....	46
<b>Fig.31.</b> personne connue.....	47
<b>Fig.32.</b> personne inconnue.....	47
<b>Fig.33.</b> Exemples des images de la base de données XM2VTS.....	48
<b>Fig.34.</b> Orientation et pose.....	49
<b>Fig.35.</b> Conditions d'illumination.....	49
<b>Fig.36.</b> Qualité de l'image.....	50

---

<b>Fig.37.</b> Expressions faciales.....	50
<b>Fig.38.</b> Présence caractéristique particulier.....	51
<b>Liste des tableaux</b>	
<b>Tableau 1.</b> Synthèse des méthodes de reconnaissance faciale.....	16
<b>Tableau 2.</b> Résultats des tests.....	53

### **I. Introduction**

Les relations sociales se basent sur plusieurs critères, notamment le visage qui reste un point essentiel à l'identification des personnes. Ces deux dernières décennies, une application de reconnaissance faciale est mise en valeur en utilisant la biométrie dans les cartes d'identités des gens.

### **II. Reconnaissance Faciale**

#### **II.1. Historique**

La reconnaissance faciale est parue pour la première fois dans les années 70 suite aux tentatives de Takeo Kanade. C'est en effet, la capacité avec laquelle une machine est capable de comprendre ce qu'elle « voit » lorsqu'on la connecte à une ou plusieurs caméras. Elle est basée sur des méthodes heuristiques en s'appuyant sur la mesure des attributs faciaux comme l'écartement des yeux, des sourcils, des lèvres, la position du menton, la forme, etc. Néanmoins, ces méthodes sont très peu fiables, car elles font de nombreuses suppositions en se plaçant dans des cas très simples (visage de face, bonnes conditions d'illuminations, etc.).

C'est qu'à partir des années 90, après l'avènement des ordinateurs, qu'un développement significatif est apparu. L'augmentation de la puissance de calcul des ordinateurs permet l'utilisation de méthodes statistiques et d'apprentissage plus complexes sur des données volumineux. Cet apport accentue la performance et distingue cette nouvelle technique de reconnaissance des précédentes méthodes de détection d'objet qui était auparavant limitées aux objets rigides et fixes et donc avec moins de variabilité. Alors qu'en 2006, le Face Recognition Grand Challenge (FRGC) a indiqué, après des tests sur des images de visage à haute résolution, des scans de visages et d'images d'iris que les nouveaux algorithmes sont 100 fois plus précis que ceux de 1995[web1].

#### **II.2. Définition**

La reconnaissance faciale comme la reconnaissance d'empreintes digitales, d'iris et vocale est un procédé biométrique qui permet de déterminer l'identité d'une personne. Elle utilise

des algorithmes pour analyser toutes les caractéristiques faciales, dans une photo ou une vidéo, telles que l'écartement des yeux, des arêtes du nez, des commissures des lèvres, des oreilles et menton

### II.3. Processus de reconnaissance des visages

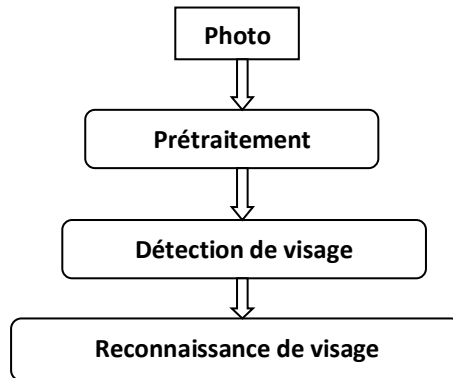


Fig. 1. Etapes de la reconnaissance du visage

Le processus complet de vérification des visages est décrit comme suit :

1. Détection de visages,
2. Extraction et normalisation des caractéristiques du visage,
3. Identification et/ou vérification.

#### II.3 .1. Détection de visage

C'est une étape par laquelle le système détecte le visage dans une image. Dans la littérature scientifique, L'image 2D acquise peut contenir à la fois le visage de la personne et éventuellement un arrière-plan.

#### II.3.2. Prétraitement : Extraction de caractéristiques du visage

L'extraction des caractéristiques des yeux, du nez et de la bouche est une étape nécessaire à la reconnaissance faciale. Dans la littérature, deux pratiques différentes sont à distinguer [Ani10].

- Approche globale : Extraction de régions entières du visage.
- Approche locale : Extraction des caractéristiques (points) particuliers de différentes régions du visage, tels que les coins des yeux, de la bouche et du nez.

Des informations complémentaires peuvent aussi être utilisées pour détecter les différents composants du visage, tels que la couleur, le mouvement ou encore la forme.

### II.3.3. La reconnaissance du visage

Une fois les caractéristiques du visage extraites, elles sont exploitées pour créer une signature numérique spécifique à une personne, qui sera stockée dans une base de données. Ainsi, La reconnaissance d'un visage requête est obtenue par l'extraction de la signature requête correspondante la plus proche dans la base de données.

## II.4 Les modes de fonctionnement

### II.4.1 Mode Surveillance

Le système de reconnaissance peut détecter automatiquement et en temps réel la forme d'un visage à l'identifier en s'appuyant sur les informations stockées dans la base de données.

### II.4.2 Mode Recherche

Le système détecte à partir d'une image de visage stable qui peut provenir de différentes sources : Appareil photo numérique, photo scanné...etc. Pour identifier des individus préenregistrer dans des bases de données

## III. Problématique

Afin de répondre à des besoins de sécurité, la reconnaissance faciale se présente comme une technique potentiellement puissante. Elle apparaît comme un domaine de recherche actif avec de nombreuses applications commerciales et d'application de la loi. Ces applications nécessitent des algorithmes robustes pour la reconnaissance du visage humain dans différentes conditions d'éclairage, expressions faciales et orientations. Ainsi, Que faut-il à la machine pour atteindre, aussi que possible, le degré de complexité de l'être humains à identifier des visages ?

## IV. Hypothèse

|| Nous pensons qu'avec un système de prédiction basé sur les chaînes de Markov cachées  
|| fournirai un taux de reconnaissance plus élevé.

## V. Objectif du travail

La reconnaissance faciale reproduit la manière dont les gens s'auto-identifient les uns des autres, c'est-à-dire en étudiant les caractéristiques de leurs visages. Nous savons tous, qu'il est quasi-impossible de simuler une telle aptitude sur une machine. Néanmoins, cette deux dernières décennies et grâce aux évolutions des sciences de technologies, notamment dans le domaine du traitement d'images, la reconnaissance faciale est devenue une technologie confirmé et précise.

Le but de notre travail est l'identification d'identité par la méthode d'apprentissage des Chaines de Markov caché. Il s'agit donc un système d'identification qui lui se charge de découvrir l'identité a priori connue d'un individu.

*L'objectif principale de ce travail est le développement d'un système d'identification de personnes par vision numérique, Le système a pour tâche d'identifier les personnes à partir des informations contenues dans une base de données préalablement construite on utilise uniquement le visage comme caractéristique discriminante.*

## VI. Présentation du mémoire

Le travail présenté dans ce mémoire s'organise autour de quatre chapitres. Le premier chapitre présente un état de l'art sur les différentes méthodes de reconnaissance de visage. Nous nous focaliserons sur les algorithmes les plus populaires et sur ceux les plus adaptés à notre contexte d'étude. Le deuxième chapitre introduit les Modèles de Markov Cachés et leurs algorithmes. Le troisième chapitre est divisé en deux sous parties à savoir : une contribution détaillée sur notre système d'apprentissage et de reconnaissance de visage à l'aide d'une méthode globale par les chaines de Markov Cachée, et une conception basée sur le langage UML2 afin de réaliser notre système de reconnaissance faciale. En dernier lieu, un dernier chapitre est consacré à l'implémentation et à la discussion des résultats. Comme convenu, tout travail, ainsi mené, s'achève par une conclusion et probablement des perspectives.

## CHAPITRE I. TECHNIQUES DE RECONNAISSANCE DE VISAGES « ÉTAT DE L'ART »

---

### I. Introduction

De nos jours, le visage peut être utilisé pour identifier une personne dans une base de données, mais il est plus communément utilisé pour vérifier l'identité. Pour la vérification des visages, ce processus est exécuté en comparant un modèle du demandeur (un ou plusieurs visages de test), avec un modèle stocké (un ou plusieurs visages de référence). De nombreuses méthodes de reconnaissance de visages ont été proposées au cours des trente dernières années. La reconnaissance faciale automatique (RFA) est un défi, elle a suscité de nombreuses recherches dans des différentes disciplines : Médecine, sciences et technologies et l'informatique (Reconnaissance des formes, vision par ordinateur...). De ce fait la littérature dans ce sens est de plus en plus diversifiée. Ce chapitre présente une vue des différentes méthodes les plus usitées en reconnaissance de visages, nous soulignons ainsi les approches globales, locales et hybrides.

### II. Les approches globales

Dans cette méthode, le visage est utilisé au grand complet comme source d'information sans segmentation de ses parties. Notons qu'ici que l'avantage réside dans la rapidité de mise en œuvre et des calculs de base. En revanche, elles sont très sensibles aux variations d'éclairément, de pose et d'expression faciale. La méthode d'analyse de données la plus convoitée est celle de l'analyse des composantes principales (ACP) communément appelée Eigenfaces.

#### II.1 Eigenfaces (Eigen = Vecteur propre)

Les Eigenfaces sont un ensemble de vecteurs propres utilisés dans le domaine de la vision artificielle afin de résoudre le problème de la reconnaissance du visage humain. Le recours à des Eigenfaces pour la reconnaissance a été développé par Sirovich et Kirby en 1987 et utilisé par Matthew Turk et Alex Pentland pour la classification de visages. Cette méthode est considérée comme le premier exemple réussi de technologie de reconnaissance faciale [Web2].

La reconnaissance de visages par Eigenface est une approche de type « image » ayant autant de dimensions que de pixels. Les caractéristiques de l'image sont extraites par une méthode mathématique de réduction de dimensionnalité basée sur un algorithme de l'analyse en composantes principales (ACP) [Man12]. Cet algorithme 2D, reposant sur les concepts de la théorie de l'information, est né des travaux de MA. Turk et al [Nic06] [Turk91]. Eigenface cherche un modèle informatique qui décrit le mieux un visage en extrayant les informations les plus pertinentes contenues dans ce visage [Hed15].

Le principe général d'Eigenface est le suivant : Étant donné un ensemble d'images de visages exemples, il s'agit tout d'abord de trouver les composantes principales de ces visages. Ceci revient à déterminer les vecteurs propres de la matrice de covariance formée par l'ensemble des images exemples. Chaque image de visage est transformée en vecteur. Chaque élément du vecteur correspond à l'intensité lumineuse d'un pixel [Phi15].

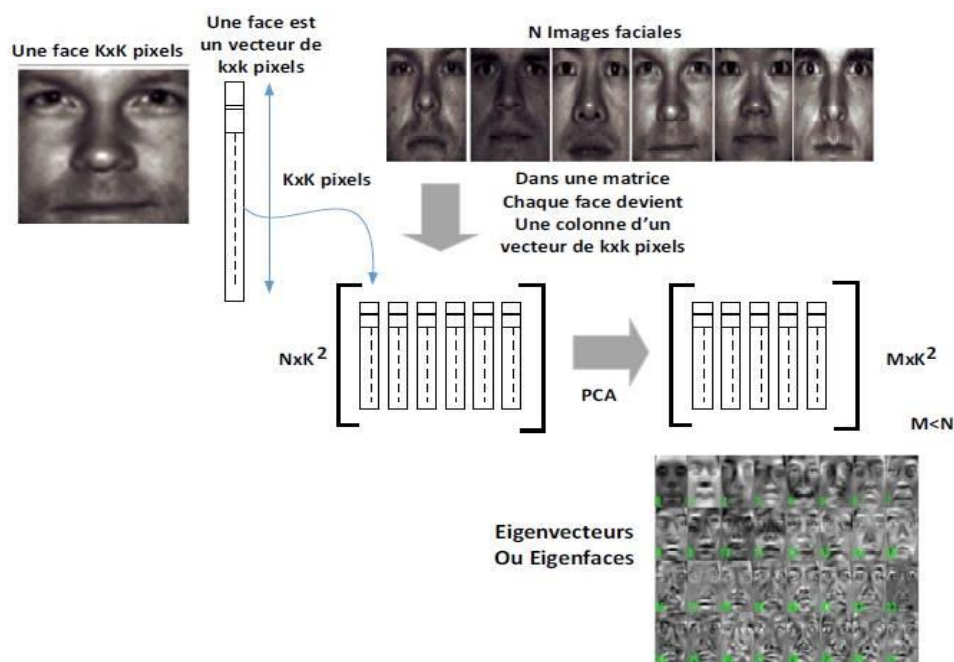


Fig.2. Reconnaissance faciale par Eigenfaces [Phi15]

## II.2 Fisherfaces

La méthode dite de Fisherface préconisée par Belhumeur et Al consiste en le principe LDA suite aux difficultés rencontrées dans le PCA quant au problème de robuste face aux variations de pose, d'illumination et d'expression. La dénomination de Fisherface et l'algorithme LDA nés des travaux de Belhumeur et Al de Yale University (USA), en

1997. L'algorithme LDA effectue une véritable séparation des classes par rapport l'algorithme PCA, organisé la base d'apprentissage des images en plusieurs classe, une classe par personne et plusieurs images par classe représente la première étape de son utilisation. L'analyse des vecteurs propres de la matrice de dispersion des données avec pour objectif de maximiser les variations entre les images des individus différents sont analysées par la LDA [Bel97]. L'application de LDA trouve des difficultés lorsque le nombre d'individus à traiter est plus faible que la résolution d'image. L'apparition des matrices de dispersions singulières lorsqu'il est difficile d'appliquer LDA. Ce problème est encore appelé Small Sample Size. Pour l'éviter, on peut utiliser tout d'abord l'ACP et ensuite l'ADL, et cette méthode est appelée Fisherfaces.



Fig.3. Reconnaissance faciale par Fisherfaces [Wih08]

### II.3 L'utilisation de Transformée en cosinus discrète DCT

La Transformée de cosinus discrète est beaucoup plus rapide tant en phase d'apprentissage qu'en phase de reconnaissance similaire aux EigenFaces, son utilisation est assez récente. Cela étant dit, chaque image de visage est représentée par un vecteur composé des premiers coefficients de la transformée. Lorsqu'un visage est présenté au module, sa transformée est calculée et un certain nombre de coefficients est retenu pour comparaison avec ceux de la banque de données. Cette dernière étape est réalisée à l'aide une distance ou avec d'autres métriques pertinentes [Rob00].

## II.4. Machine à Vecteurs de Support (SVM)

L'idée essentielle de cette technique 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 [Guo00]. Le calcul de la fonction de classement séparant les classes est assuré par la technique de construction de l'hyperplan optimal.

## II.5. Les réseaux de neurones (RNA)

Le professeur Tom Mitchell a conçu une architecture énorme, plus de 20 NN pour reconnaître 20 personnes avec son équipe de laboratoire, il obtient des résultats favorables [Tom 97]. L'utilisation des réseaux de neurones comme engin d'apprentissage et de reconnaissance représente la technique de Tom Mitchell et ces collaborateurs. La source d'entrée des réseaux est la technique pour débiter une image brute de dimension fixe. Les dimensions doivent être établies au préalable car le nombre de neurones sur la couche d'entrée en dépend. Plus les dimensions de l'image sont élevées plus la complexité et le temps d'apprentissage augmentent.

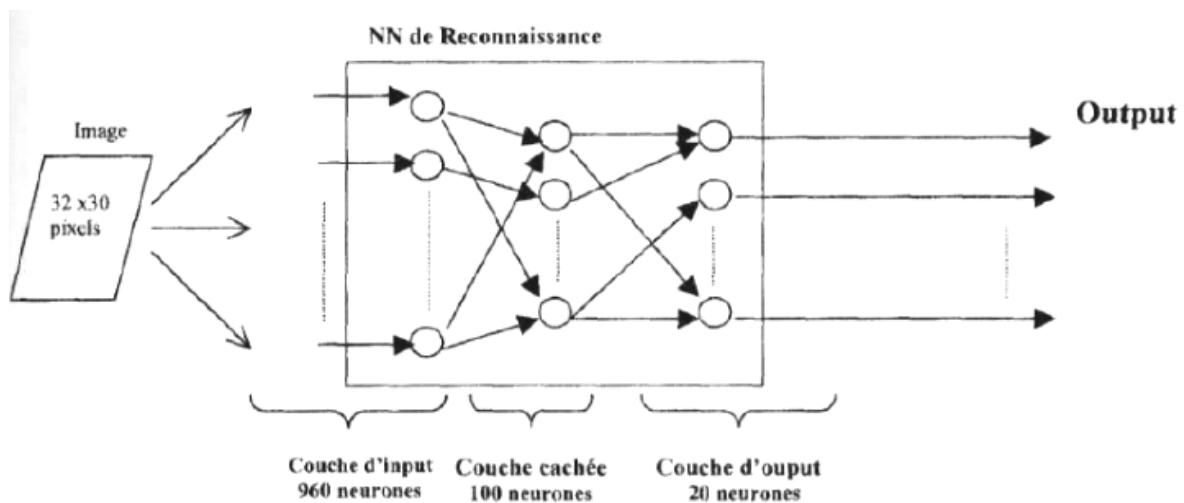


Fig.4. Reconnaissance faciale par réseaux de neurones [Man13]

## III. Les approches locales

On les appelle aussi les méthodes à traits, à caractéristiques locales ou les méthodes analytiques. Ce type consiste à appliquer des transformations en des endroits spécifiques de

l'image, le plus souvent autour de points caractéristiques (coins des yeux, de la bouche, le nez, ...). Elles nécessitent donc une connaissance a priori sur les images [Min02].

### III.1 Eigen Objects (EO)

Les EigenObjects visent des parties bien précises du visage. La réalisation de l'apprentissage commence d'abord par la procédure à une ACP des yeux contenus dans la banque de visages. Ce qui va servir au processus de reconnaissance [Ahm15]. L'élément essentiel de ces méthodes repose sur la réalisation d'une analyse en composantes principales (ACP) sur une série d'images d'objets à détecter. La réduction de dimensionnalité fournit les premiers vecteurs propres (d'où le terme eigen) qui représente les grandes différences entre les objets d'intérêt. La technique des EigenObjects utilise certaines hypothèses de base telles que les positions relatives des différentes composantes du visage. On peut donc combiner les deux méthodes pour diminuer le temps de recherche et raffiner les résultats [Ale03].

### III.2 Elastic Bunch Graph Matching

L'algorithme EBGM de Wiskott et Alen localise des points caractéristiques à partir d'une image de visage manuellement ou automatiquement à l'aide d'un algorithme. Un treillis élastique virtuel sur l'image de visage est appliqué. Chaque point représente un nœud labellisé associé à un jeu de coefficients d'ondelettes complexes de Gabor, appelés Jet<sup>1</sup> afin d'effectuer une reconnaissance d'une image test, la technique consiste à faire une mesure de similarité entre les différents Jets et les longueurs des segments du treillis de deux images [Tho04]. La caractéristique de l'EBGM est une grande robustesse aux changements d'éclairage, de pose et d'expression faciale.

---

<sup>1</sup> Un Jet peut être écrit comme un ensemble de coefficients complexes  $\mathcal{J}_i = a_i \exp(i\phi_i)$  avec une amplitude  $a_j$  qui varie peu avec la position, et une phase  $\phi_j$ , dont la variation en rotation est plus importante.

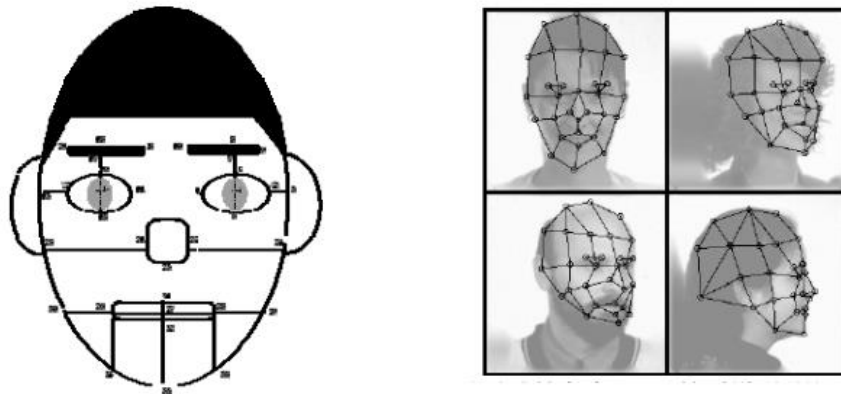


Fig.5. Reconnaissance faciale par Elastic Buch Graph Matching [Shr14]

### III.3 Couleurs

Les images en couleur diffèrent de celles en niveau de gris, cet avantage peut être exploité pour tirer des zones d'intérêt en se basant sur la couleur de la peau. On trouve d'ailleurs beaucoup de recherches sur la détection de la peau, car elle représente un espace de couleur un peu unique. En utilisant les frontières de cette région comme valeurs de seuillage sur une image, il est possible d'extraire les pixels dont la couleur peut s'apparenter à celle de la peau [Gar99]. Plusieurs modèles de couleurs pouvant s'appliquer à la détection de la peau cette variabilité dépend de l'espace de couleur adopté pour la représentation de la chrominance des pixels les plus utilisés sont : l'espace RGB<sup>2</sup>, HSV<sup>3</sup>, YCbCr<sup>4</sup>.

<sup>2</sup> RGB en anglais « Red, Green, Blue » systèmes de codage informatique des couleurs, chaque paramètre (rouge, vert et bleu) définit l'intensité de la couleur comme un nombre entier compris entre 0 et 255.

<sup>3</sup> Le système HSV (Hue, Saturation, Value) est un espace colorimétrique, défini en fonction de ses trois composantes teinte : varie entre 0 et 360°, saturation : varie entre 0 et 100 %, valeur: varie entre 0 et 100%.

<sup>4</sup> Le système Y'CbCr est utilisé pour les images JPEG avec trois composantes Y (luminance) et les différences de couleur bleu (Cb) et Rouge (Cr).

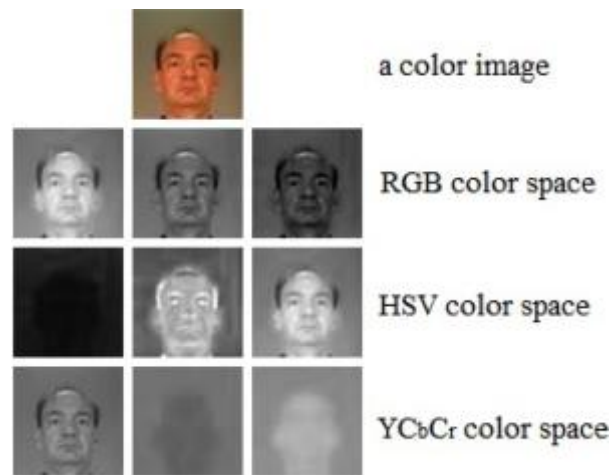


Fig.6. Reconnaissance faciale par la couleur de peau [Qia17]

### III.4 L'appariement du gabarit (Template Matching)

Elle consiste en effet à comparer l'intensité des pixels entre un gabarit prédéfini et plusieurs sous régions de l'image à analyser. Ce processus effectue plusieurs balayages couvrant toute la superficie de l'image. Les régions de doute de la présence de visages seront donc facilement identifiées par des minimums de distance entre le gabarit et l'image sous-jacente [Ale03].

Certains travaux [Min02] ont utilisé une détection des caractéristiques du visage à l'aide de gabarits plus spécialisés (p. ex. : yeux, bouche, etc.).



Fig.7. Reconnaissance faciale par Template Matching [Ank11]

### III.5 Modèle de Markov caché

Les modèles de Markov cachés (HMM) est une approche statistique utilisés pour la détection et la reconnaissance du visage. Les premiers travaux relèvent des auteurs Samaria et Yong. Ils ont présentés le HMM comme un mécanisme assez robuste pour faire la reconnaissance faciale [Fer94]. Le HMM utilisé était un seul HMM gauche à droite comme le montre la figure8 où chaque état est modélisé comme une région spécifique d'un visage.

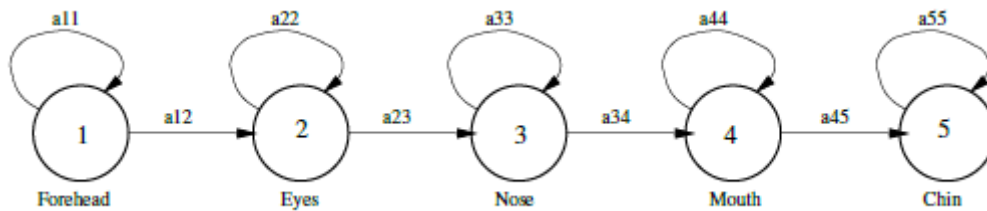


Fig.8. HMM de gauche à droite [Fer94]

Samaria a étendu le modèle HMM classique de gauche à droite à un modèle pseudo-bidimensionnel (pseudo-2D) figure9. Le HMM consiste en un ensemble de super-états de niveau supérieur, chacun contenant un ensemble d'états incorporés, utilisés pour modéliser les données bidimensionnelles dans une direction où une autre inversée par autre HMM intégrés. Ce modèle est approprié pour les images de visage car il exploite les structures physiques 2D d'un visage, à savoir qu'un visage conserve la même structure d'états de haut en bas - front, yeux, nez, bouche, menton, et même gauche à bonne structure d'états à l'intérieur de chacun de ces super états [Sam&You94]. Cette approche a été testée sur la base de données ORL ( Olivetti Research Laboratory).



Fig.9. Structure d'un HMM 2D [Sam&You94]

Une version améliorée basé sur les HMM a été proposé par Eickeler et al[Eic99]. Ces derniers profitent de l'algorithme de format JPEG pour reconnaître les visages directement à partir des données compressées. C'est-à-dire il utilise directement les fonctionnalités DCT fournies par le standard JPEG, sans qu'il soit nécessaire de décompresser complètement l'image avant la reconnaissance. Les taux de reconnaissance sur la base de données des visages Olivetti Research Laboratory (ORL) sont de 100% pour les images originales et de 99,5% pour la reconnaissance de domaine comprimée JPEG.

Othman et Aboulnasr [Oth00] ont proposé un système HMM 2D (LC2D HMM Low-Complexity) de faible complexité pour la reconnaissance faciale. Les auteurs sollicitent une réduction de complexité de calcul de  $N^4$  pour un HMM 2D complètement connecté à  $2N^2$  pour le HMM LC2D, où  $N$  est le nombre d'états. Bien que la précision du système ne soit pas meilleure que d'autres approches, ces auteurs affirment que la complexité de calcul impliquée est quelque peu inférieure à celle requise pour un HMM 1D et significativement inférieure à celle du HMM P2D.

Le HMM LC2D repose sur deux hypothèses clés:

- 1- L'état actif du bloc d'observation  $B_{k,l}$  dépend uniquement des voisins verticaux et horizontaux immédiats,  $B_{k-1,l}$  et  $B_{k,l-1}$ <sup>5</sup>;
- 2- Les états actifs des deux blocs d'observation dans les emplacements de voisinage anti-diagonaux,  $B_{k-1,l}$  et  $B_{k,l-1}$  sont statistiquement indépendants compte tenu de l'état actuel. Cette hypothèse permet de séparer la matrice de transition d'état 3D en deux matrices de transition 2D distinctes, pour les transitions horizontales et verticales. Cela diminue considérablement la complexité du modèle [Oth00].

---

<sup>5</sup> D'un point de vue mathématique, cette hypothèse est équivalente à un modèle de Markov de second ordre, nécessitant une matrice de transition 3D.

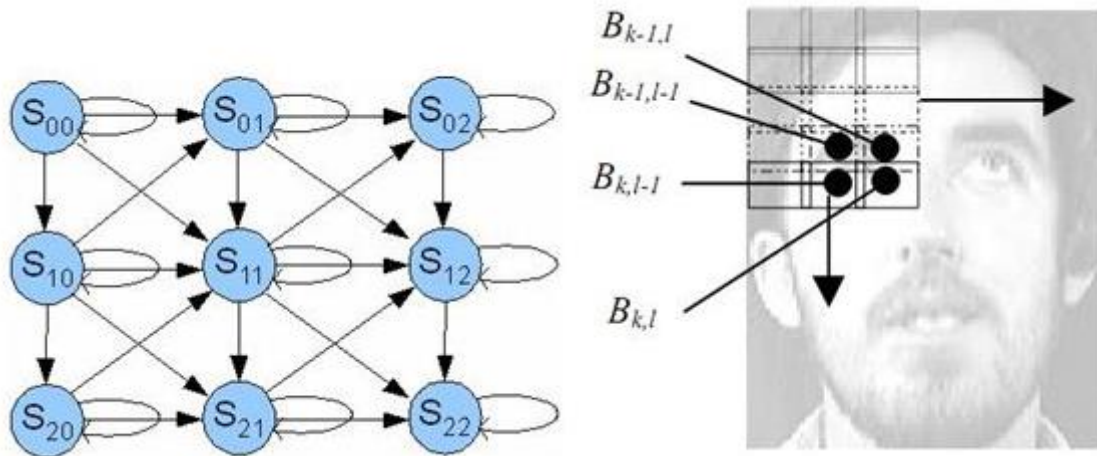


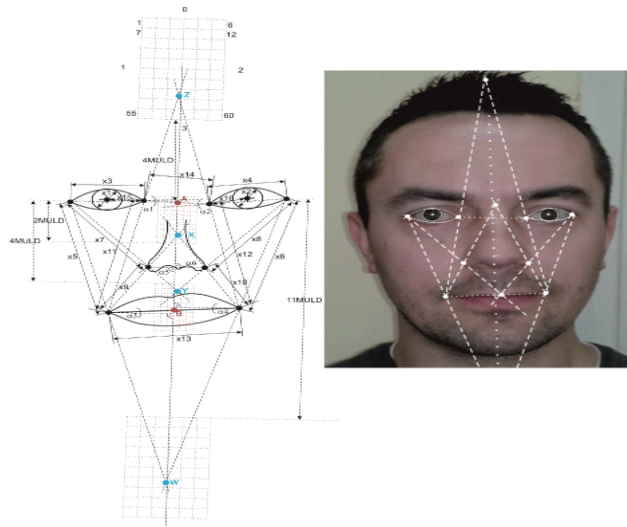
Fig. 10. Topologie de modèle [Oth00]

Les travaux Yujian [Yuj07] fournissent plusieurs nouvelles formules analytiques pour résoudre les problèmes de base de HMM 2D. La complexité de ces calculs est la même que celle d'un HMM 1D pour les cas où les nombres de lignes ou de colonnes sont une petite constante. Ses travaux simplifient l'application d'un HMM 2D complet au problème de reconnaissance faciale sans pour autant avoir appliqué ses résultats spécifiquement à la reconnaissance faciale.

D'autres travaux décrits dans [Pet11] où ils s'intéressent aux systèmes embarqués pour la reconnaissance faciale. Pour cela, ils ont testé le système EHMM pour différentes tailles du modèle, un nombre différent de gaussiennes, la taille de l'image, les caractéristiques et le nombre d'images par personne utilisées pour l'apprentissage. Les résultats de reconnaissance obtenus pour une image (32x32) par personne ne sont que 58% alors que pour 5 images par personne le taux arrive au seuil de 82%.

Mariusz et Bobulski optent pour une approche markovienne pour la vérification d'identité basée sur l'asymétrie faciale d'où une nouvelle méthode de vérification basée sur des points caractéristiques désignés [Mar17].

Les informations codées désignent ces points en tant que vecteurs d'observation et reconnaissance à l'aide de modèles de Markov cachés. La distance d'angles et de points d'intersection déterminés des lignes auxiliaires représente la construction de vecteurs d'observation de l'asymétrie faciale.



**Fig. 11. Diagramme de création de modèle d'asymétrie faciale [Mar17]**

Les modèles ont une meilleure gestion si les points de base sont correctement déterminés, le modèle que donnent de faux résultats lorsqu'il existe une grande de base de données d'images

Récemment, les modèles de Markov cachés ont été utilisés dans l'analyser des données de mouvements oculaires de la reconnaissance faciale [Tim14].

Les travaux Tim Chuk et Al ont opérés une tâche de reconnaissance faciale avec des participants asiatiques et modélisés le modèle de mouvements oculaires de chaque participant avec un HMM, Par regroupement de ces HMM, ils ont montrés que les mouvements oculaires des participants pouvaient être catégorisés en modèles holistiques ou analytiques, démontrant des différences individuelles significatives même au sein d'une même culture.

Méthodes	• Avantages	• Inconvénients
Eigenface	<ul style="list-style-type: none"> <li>Aucune connaissance de la géométrie du visage n'est requise</li> </ul>	<ul style="list-style-type: none"> <li>Méthode est très sensible à l'échelle</li> </ul>
Fisherfaces	<ul style="list-style-type: none"> <li>Fisherface est similaire à Eigenface, mais avec une amélioration de la meilleure classification des différentes classes d'image.</li> <li>Avoir une meilleure précision dans l'expression du visage que l'approche Eigenface</li> </ul>	<ul style="list-style-type: none"> <li>Fisherface est plus complexe qu'Eigenface</li> <li>Nécessite un stockage plus important du visage et plus de temps de traitement en reconnaissance</li> </ul>
Machine à Vecteurs de Support (SVM)	<ul style="list-style-type: none"> <li>Basé sur une théorie forte et agréable</li> </ul>	<ul style="list-style-type: none"> <li>Tend à être coûteux en mémoire et en temps de calcul</li> </ul>
Eigen objects (EO)	<ul style="list-style-type: none"> <li>Mesure de similarité</li> <li>Moins sensible à l'éclairage</li> </ul>	<ul style="list-style-type: none"> <li>Idem à l'Eigenface</li> </ul>
Elastic Bunch Graph Matching(EBGM)	<ul style="list-style-type: none"> <li>Une grande robustesse aux changements d'éclairage, de pose et d'expression faciale</li> </ul>	<ul style="list-style-type: none"> <li>Faible précision (où les caractéristiques clés les yeux, le nez, la bouche sont occultées n'est pas claire)</li> </ul>
Template Matching	<ul style="list-style-type: none"> <li>Conceptuellement simple</li> <li>Mesure de similarité</li> </ul>	<ul style="list-style-type: none"> <li>Faible précision (les occultations et les conditions d'illumination)</li> </ul>
Couleur	<ul style="list-style-type: none"> <li>Détection de la peau efficace</li> </ul>	<ul style="list-style-type: none"> <li>Détection des yeux peu robuste.</li> <li>Conflits avec l'arrière plan.</li> </ul>
Réseaux de Neurones	<ul style="list-style-type: none"> <li>Apprentissage automatique</li> </ul>	<ul style="list-style-type: none"> <li>Nécessite un temps assez important dans le cas d'apprentissage</li> </ul>
Modèle de Markov caché	<ul style="list-style-type: none"> <li>L'une des propriétés importantes des modèles de Markov cachés est la possibilité de modéliser des dépendances séquentielles</li> </ul>	<ul style="list-style-type: none"> <li>complexité de calcul pour un HMM 2D</li> </ul>

Tableau 1. Synthèse des méthodes de reconnaissance faciale

## Conclusion

Ce chapitre explore les méthodes et les techniques existantes de la reconnaissance faciale. Nous avons présenté une revue des techniques les plus intéressantes et les plus largement utilisées dans ce domaine, pour en terminer avec une littérature sur les différentes recherches reposant sur les modèles HMM pour l'identification du visage.

Il est clair que différentes techniques équilibrent certains compromis entre la complexité de calcul et la rapidité de la reconnaissance.

## CHAPITRE II. MODELE DE MARKOV CACHEE « DESCRIPTION »

---

### I. Introduction

Les modèles de Markov cachés sont des outils statistiques permettant de modéliser des phénomènes stochastiques. Ces modèles sont utilisés dans de nombreux domaines tels que la reconnaissance et la synthèse de la parole, la biologie, l'indexation de documents, la reconnaissance d'images, la prédiction de séries temporelles, . . . Pour pouvoir utiliser ces modèles efficacement, il est nécessaire d'en connaître les principes.

Ce chapitre présente les caractéristiques fondamentales de la théorie markovienne et leur utilisation dans la reconnaissance faciale.

### II. Modèle de Markov

Le modèle de Markov est représenté par la Chaîne de Markov, une chaîne de Markov est de manière générale un processus de Markov à temps discret et à espace d'états discret. En mathématiques, un processus de Markov est un processus stochastique possédant la propriété de Markov.

#### II.1 Les concepts de chaîne de Markov

Une chaîne de Markov est une suite de variables aléatoires  $(X_n, n \in \mathbb{N})$  qui permet de modéliser l'évolution dynamique d'un système aléatoire :  $X_n$  représente l'état du système à l'instant  $n$ . La propriété fondamentale des chaînes de Markov, dite propriété de Markov, est que son évolution future ne dépend du passé qu'au travers de sa valeur actuelle.

##### II.1.1 Processus stochastiques

On appelle processus stochastique tout phénomène d'évolution temporelle dont l'analyse peut être soumise au calcul des probabilités. Du point de vue de l'observation, un processus stochastique est constitué par l'ensemble de ses réalisations. Une réalisation est obtenue par l'expérience qui consiste à enregistrer une suite d'événements au cours du temps. Le caractère aléatoire de l'évolution se montre par le fait que la répétition de l'expérience conduit à une autre séquence temporelle [Phi06].

### II.1.2 Processus de Markov

Un processus de Markov est un processus dont l'évolution future  $\{X_s : s > t\}$  ne dépend de son passé qu'à travers son état à l'instant  $t$  [Bou03]:

$$\forall s > t, p(X_s | X_r : r \leq t) = p(X_s | X_t) \quad (1)$$

Où  $P(X_s | X_t)$  désigne la loi de probabilité de  $X_s$  sachant  $X_t$ .

L'indice  $t$  est souvent interprété comme le temps.

Cette définition signifie que, pour le futur, l'histoire du processus jusqu'à l'instant  $t$  est entièrement résumée par son état à l'instant  $t$ ; ou encore que le présent étant connu, le futur est indépendant du passé.

### II.1.3 Chaîne de Markov à temps discret

La suite  $\{X_n\}_{n \in \mathbb{N}}$  est une chaîne de Markov à temps discret si et seulement si :

$$P(X_n=j | X_{n-1}=i_{n-1}, X_{n-2}=i_{n-2}, \dots, X_0=i_0) = P(X_n=j | X_{n-1}=i_{n-1}) \quad (2)$$

La probabilité pour que la chaîne soit dans un certain état à la  $n^{\text{ième}}$  étape du processus ne dépende donc que de l'état du processus à l'étape précédente (la  $n-1^{\text{ième}}$ ) et pas des états dans lequel il se trouvait aux étapes antérieures

On peut alors définir la probabilité de transition d'un état  $i$  vers un état  $j$  notée  $a_{ij}$  (appelée Densités de transitions):

$$a_{ij} = P(X_n=j | X_{n-1}=i) \quad \forall n \in \mathbb{N}. \quad (3)$$

$$\text{ou } \sum a_{ij} = 1.$$

On définit alors la matrice de transition  $A$ :

$$A = \begin{pmatrix} a_{00} & a_{01} & \dots \\ \vdots & \ddots & \vdots \\ \vdots & \vdots & a_{nn} \end{pmatrix} \quad (4)$$

Prenons l'exemple d'une station météo qui veut modéliser l'évolution du temps

Regardons trois aspects du temps correspondant à trois états :

État 1=S1 : Pluie,

État 2=S2 : Nuage,

État 3=S3 : Soleil.

Chacun des états peut être suivi de l'un des trois autres, certaines transitions étant plus probables que d'autres. Donner un modèle M de l'évolution du temps revient à déterminer ces transitions (Figure 12). Nous définissons alors une matrice A de transitions, où les termes  $a_{ij}$  correspondent à la probabilité d'évoluer de l'état  $S_i$  vers l'état  $S_j$   $a_{ij} = P(S_i/S_j)$  et les probabilités initiales  $\pi = P(S_i)$ .

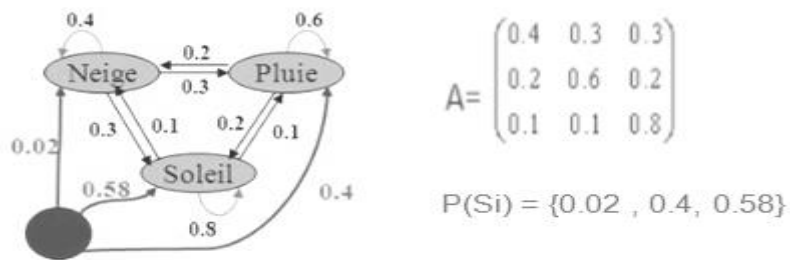


Fig. 12. Exemple de modélisation de Chaîne de Markov

Donner une prédiction à partir de ce modèle, c'est donner la probabilité que la succession d'états souhaités se réalise. Si l'on observe le temps «pluie» (état  $s_1$ ) le Samedi, la probabilité que le temps du reste de la semaine soit «pluie, pluie, pluie, pluie, soleil, nuage» se calcule de la manière suivante : Soit  $O = (s_1, s_1, s_1, s_1, s_1, s_3, s_2)$  une suite d'observations. La probabilité que cette observation se réalise est :

$$\begin{aligned} P(O/M) &= P(s_1, s_1, s_1, s_1, s_1, s_3, s_2) \\ &= P(s_1, s_1, s_1, s_1, s_1, s_3) \times P(s_2/s_1, s_1, s_1, s_1, s_1, s_3) \\ &= \dots \\ &= P(s_1) \times P(s_1/s_1) \times P(s_1/s_1, s_1) \times P(s_1/s_1, s_1, s_1) \times P(s_1/s_1, s_1, s_1, s_1) \\ &\quad \times P(s_3/s_1, s_1, s_1, s_1, s_1) \times P(s_2/s_1, s_1, s_1, s_1, s_1, s_3) \end{aligned}$$

Grâce à l'hypothèse de Markov d'ordre 1, l'observation courante ne dépend que de l'état précédent :

$$P(s_t/s_{t-1}, s_{t-2}, \dots, s_0) = P(s_t/s_{t-1})$$

L'égalité précédente devient alors :

$$P(O/M) = P(s_1) \times P(s_1/s_1) \times P(s_1/s_1) \times P(s_1/s_1) \times P(s_1/s_1) \times P(s_3/s_1) \times P(s_2/s_3)$$

### III. Chaînes de Markov Cachées

L'étude probabiliste des phénomènes s'introduit naturellement lorsqu'il existe une incertitude sur la mesure décrivant un phénomène. Considérons deux phénomènes « mesurés » par deux réels  $x$  et  $y$ . Lorsqu'on cherche à étudier des liens entre les deux phénomènes il existe, en dehors de la théorie des probabilités, deux possibilités : soit un lien déterministe  $y=f(x)$ , soit aucun lien. Le calcul des probabilités permet d'introduire une infinité de « liens intermédiaires » : les deux phénomènes peuvent être plus au moins liés. Lorsqu'on « observe »  $y$ , on dispose sur  $y$  d'une certaine information, cette information est modélisée par une « mesure de probabilité » notée  $P_{Y|x}$  : pour tout  $A \subset \mathbb{R}$ , est un nombre dans  $[0, 1]$  donnant la « probabilité » pour que  $y$  soit dans  $A$ ,  $y$  est ainsi considéré comme réalisation d'une variable aléatoire  $Y$  et  $P_{Y|x}$  est la loi de  $Y$  conditionnelle à  $x$ .

D'une façon générale  $X$  contient l'information que l'on recherche mais n'est pas directement observable (on dit généralement qu'elle est « cachée »). On observe, ou mesure,  $Y = y$  et on cherche à retrouver, ou à « estimer », la réalisation cachée  $x$ . Considérons, à titre d'exemple, le problème suivant : on cherche à savoir, à partir d'une image, si un certain pixel de l'image représente de la forêt ou de l'eau.  $X$  prend ainsi ses valeurs dans un ensemble de deux éléments  $\Omega = \{s_1 = \text{"eau"}, s_2 = \text{"forêt"}\}$  et l'observation  $Y = y$  est donnée par un nombre représentant un niveau de gris, que l'image a été prise dans une région où il y a trois fois plus d'eau que de forêt, on posera  $P_X(s_1) = 0.75$  et  $P_X(s_2) = 0.25$ . Les deux lois conditionnelles  $P_{Y|s_1}$  et  $P_{Y|s_2}$  modélisent plusieurs phénomènes déferents [Sté12].

#### III.1 Définition

Une chaîne de Markov cachée est un processus à temps discret ou encore composé de deux processus  $X=(X)_n \ n \in \mathbb{N}$  et  $Y=(Y)_n \ n \in \mathbb{N}$ . Le processus  $X$  est une chaîne de Markov dont les réalisations sont inobservables et le processus  $Y$  est réel (chaque  $Y_n$  prend ses valeurs dans  $\mathbb{R}$ ) [Ouh12].

Formellement, un HMM est défini par:

- L'alphabet  $X_s \in \{s_1, \dots, s_k\}$  des états cachés de la chaîne de Markov
- La matrice de transition  $A = \{a_{ij} = P(x_j | x_i)\}$
- Les probabilités de départ  $\Pi = \{\pi_i = P(x_i)\}$
- L'alphabet  $\Sigma = \{y_1, \dots, y_k\}$  des symboles émis par les  $x_i$  pour un HMM discret
- Les probabilités d'émission  $Y = \{f_i(y_k) = P(y_k | x_i)\}$

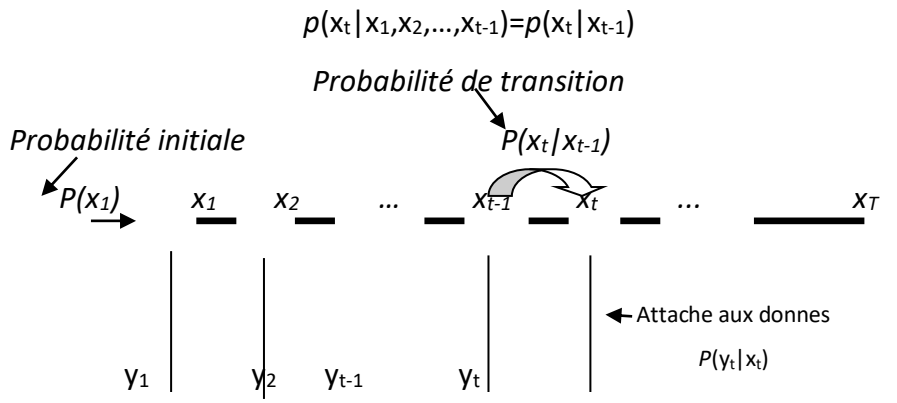


Fig. 13. Chaîne de Markov Cachée d'ordre 1

La figure 13 présente le graphe d'une chaîne de Markov d'ordre 1: Y représente les observations à partir desquelles on recherche les états cachés X.

### III.2 Théorie bayésienne

La théorie bayésienne est basée sur la spécification de la distribution a posteriori

$p(X=x|Y=y)$ . La règle de Bayes permet de déterminer la probabilité a posteriori des états connaissant le champ des observations [Sté08]:

$$p(X = x | Y = y) = \frac{p(X = x, Y = y)}{p(Y = y)} = \frac{p(Y = y | X = x)p(X = x)}{p(Y = y)} \quad (5)$$

Où  $p(Y = y | X = x)$  est la vraisemblance des observations conditionnellement aux informations cachées,  $p(X = x)$  est la loi à priori sur les états et  $p(Y = y)$  une constante de normalisation.

#### III.2.1 La loi de X a priori

La loi de X est déterminée par la connaissance, d'une part de la probabilité initiale d'appartenance à une classe  $\pi_i$ , et d'autre part des probabilités de transition d'une classe à l'autre  $a_{ij}$  [Pie06] :

$$\pi_i = p(x_1 = s_i) \quad 1 \leq i \leq K \quad (6)$$

Nous supposons pour la suite :

$$a_{ij} = p(x_n = s_i, x_{n+1} = s_j) \quad 1 \leq i \leq K, 1 \leq j \leq K \quad (7)$$

### III.2.2 Loi de (X, Y)

On définit maintenant les lois de Y sachant X.

$$p(x, y) = p(x) \prod_{n=1}^s p(y_n/x_n). \forall n \in S \quad (8)$$

$$= p(x_1) p(y_1/x_1) \prod_{n=2}^s p(x_n/x_{n-1}) p(y_n/x_n). \forall n \in S \quad (9)$$

Soit finalement :

$$p(x, y) = \pi(x_1) f_{x1}(y_1) \prod_{n=2}^s t(x_{n-1}, x_n) f_{xn}. \forall n \in S \quad (10)$$

Les densités conditionnelles  $f_{xn}(y_n)$  sont également appelées lois d'émission [Sté12]

$$f_{xn}(y_n) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(y-\mu)^2}{2\sigma^2}} \quad (11)$$

## III.3 Prérequis Statistiques

### III.3.1 Moyenne

On considère un vecteur X représentant une distribution de n données :  $X = [X_1, X_2, \dots, X_i, \dots, X_n]$ . La moyenne  $\mu$  de cette distribution s'écrit :

$$\mu = \sum_{i=1}^n \frac{x_i}{n} \quad (12)$$

### III.3.2 Ecart-type

L'écart-type est une mesure de la dispersion d'un ensemble de données. D'un point de vue qualitatif, l'écart-type caractérise la largeur d'une distribution de données en mesurant la dispersion autour de la moyenne.

La formule de l'écart-type est :

$$\sigma = \sqrt{\sum_{i=1}^n \frac{(X_i - \mu)^2}{n-1}} \quad (13)$$

**IV. Problèmes fondamentaux**

Classiquement, les CMC introduisent trois problèmes fondamentaux [Moh99] :

**IV.1 Premier problème: L'évaluation**

Comment évaluer la vraisemblance d'une observation par rapport à un modèle c'est-à-dire comment calculer  $P(O = o|\lambda)$ ? Ce problème est efficacement résolu par l'algorithme de Forward. La variable Forward comme la probabilité d'observer la suite  $o = (o_1, o_2, \dots, o_t)$  à l'instant  $t$  en étant dans l'état  $s_i$  et connaissant la chaîne de Markov cachée  $\lambda$ .

$$\alpha_t(i) = P(o_1 \dots o_t; q_t = s_i | \lambda) \tag{14}$$

A l'instant initial ( $t = 1$ ), la variable Forward  $\alpha_1(i)$  est la probabilité jointe d'observer le premier symbole de l'observation ( $o_1$ ) et de se trouver dans l'état  $s_i$ . On a donc

$$\alpha_1(i) = \pi_i \times b_i(k_1) \tag{15}$$

avec  $o_1 = v_{k_1}$ .

Puis, une phase d'induction calcule  $\alpha_t(j)$  pour  $t = 2 \dots T$ .

$$\alpha_t(j) = \left[ \sum_{i=1}^N \alpha_{t-1}(i) \times a_{ij} \right] \times b_j(k) \quad \begin{cases} 2 \leq t \leq T \\ 1 \leq j \leq N \\ o_t = v_k \end{cases} \tag{16}$$

La variable  $\alpha_T(j)$  est par définition la probabilité d'avoir toute l'observation en terminant dans l'état  $s_j$ . La sommation de cette variable sur l'ensemble des états finaux possibles donne  $P(O = o_1)$ .

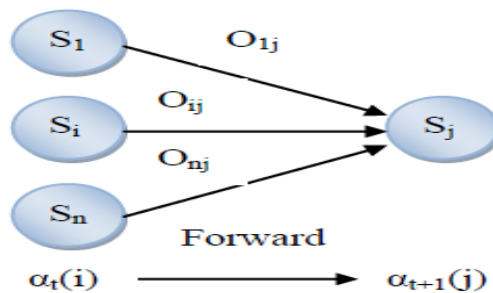


Fig.14. Séquence d'opérations pour forward [Bel15]

**Algorithme de Forward** [Deq05] :

**Pour** i=1 à N **Faire**

$$\alpha_1(i) = \pi_i b_i(o_1)$$

**FinPour**

**Pour** t=1 à T-1 **Faire**

**Pour** j=1 à N **Faire**

$$\alpha_{t+1}(j) = (\sum_{i=1}^N \alpha_t(i) a_{ij}) b_j(o_{t+1})$$

**FinPour**

**FinPour**

$$P(o/\lambda) = \sum_{i=1}^N \alpha_T(i)$$

On peut considérer le problème d'une façon analogue. On obtient alors l'algorithme Backward, où le terme générique est  $\beta_t(j) = P(o_{t+1} \dots o_T | Q_t = s_j)$ , la probabilité de générer l'observation  $(o_{t+1} \dots o_T)$  et d'être dans l'état  $s_j$  à l'instant t.

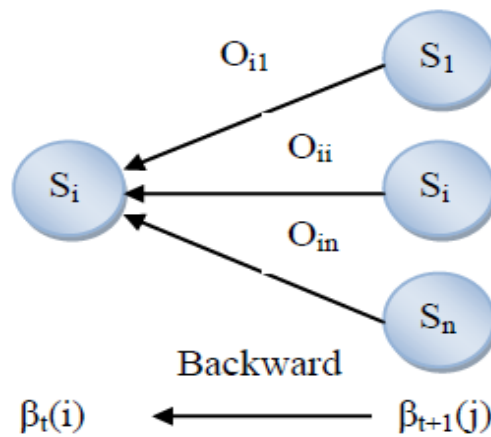


Fig.15. Séquence d'opérations pour Backward [Bel15]

**Algorithme de Backward** [Deq05] :

**Pour**  $i=1$  à  $N$  Faire

$$\beta_T(i) = 1$$

**FinPour**

**Pour**  $t=T-1$  à  $1$  Faire

**Pour**  $i=1$  à  $N$  Faire

$$\beta_t(i) = \sum_{j=1}^N a_{ij} \beta_{t+1}(j) b_j(o_{t+1})$$

**FinPour**

**FinPour**

$$P(o/\lambda) = \sum_{i=1}^N \pi_i b_i(o_1) \beta_1(i)$$

#### IV.2 Deuxième problème : Quelle est la X la plus probable ?

Le problème posé ici est de trouver, étant donné une séquence de symboles  $O = o_1 \dots o_T$  et un CMC la séquence d'états du CMC qui a la probabilité maximale de générer  $O$ . Ce qui nous préoccupe n'est pas la valeur de la probabilité maximale mais le chemin, appelé chemin de Viterbi, qui permet de générer la séquence  $O$  avec cette probabilité. De manière similaire à l'approche utilisée pour le calcul de  $P(O | X)$ , l'approche directe pour résoudre ce problème consiste à calculer la probabilité en de génération suivant tous les chemins possibles et de choisir parmi ces chemins celui qui a la probabilité la plus élevée. L'algorithme de Viterbi est un algorithme de programmation dynamique très similaire à l'algorithme forward et qui permet de résoudre efficacement ce problème [Bel15].

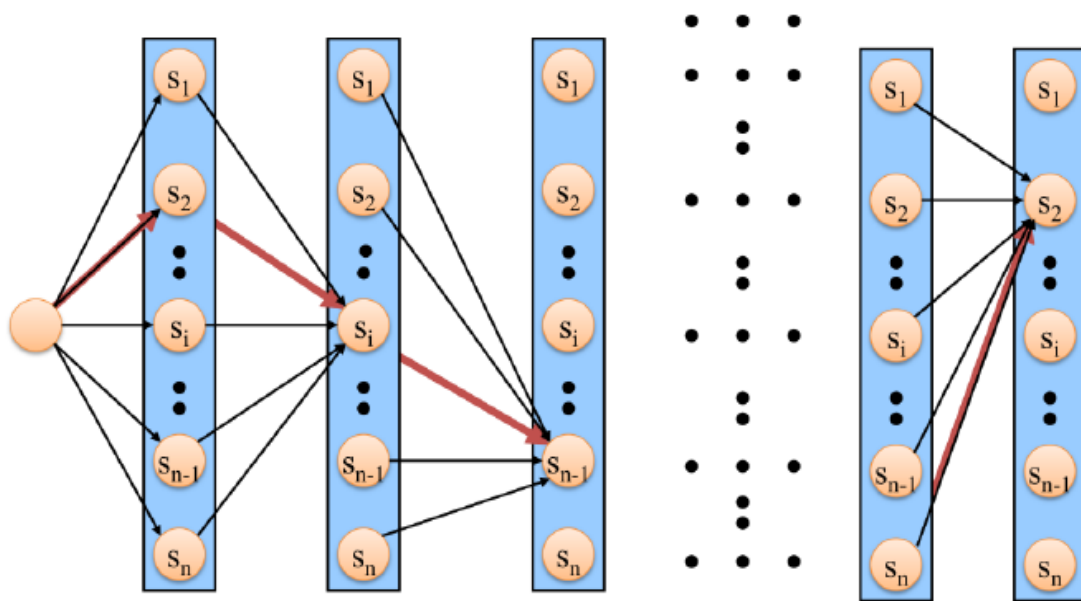


Fig.16.Principe de fonctionnement de l’algorithme Viterbi [Bel15]

L’algorithme de viterbi [You16]:

**Initialisation :**

$$\text{Pour } t=1 \text{ et } 1 \leq i \leq \text{nombre d'états } m, \quad \alpha_1(i) = \pi_i * b_i(o_1)$$

$$\psi_1(i) = 0$$

**Récurrance**

$$\text{Pour } t=2, \dots, n, \quad \alpha_{t+1}(i) = \max_{j=1..m} [\alpha_t(j) * a_{ji}] * b_i(o_{t+1})$$

$$\text{Et } 1 \leq i \leq m, \quad \psi_t(i) = \operatorname{argmax}_{j=1..m} (\alpha_{t-1}(j) * a_{ji})$$

**Terminaison :**

$$S(n) = \operatorname{argmax}_i \alpha_T(i)$$

**Retour en arrière :**

$$\text{Pour } t=n-1, \dots, 1, \quad s(t) = \psi_{t+1}(s(t+1))$$

### IV.3 Troisième problème : Apprentissage

Comment ajuster les coefficients des matrices A, B, Π pour qu'un modèle reconnaisse mieux une observation ? Pour résoudre ce problème efficacement, on utilise généralement l'algorithme de Baum Welch.

L'algorithme de Baum-Welch réestime les valeurs des matrices A, B et Π de façon à maximiser la vraisemblance de l'observation O:  $P(O=o)$ . On obtient après exécution de l'algorithme la CMC  $\lambda^*$  telle que [Flo07]:

$$\lambda^* = \operatorname{argmax}_{\lambda} P(O = o|\lambda) \quad (17)$$

Les estimations peuvent donc logiquement se concevoir :

$$\pi_i = \text{probabilité d'être dans l'état } S \text{ à l'instant } t = 1$$

$$\bar{a}_{ij} = \frac{\text{nombre de transitions de l'état } S_i \text{ vers } S_j}{\text{nombre de fois où l'on quitte } S_i}$$

$$\bar{b}_{ij}(k) = \frac{\text{nombre de fois où l'on est dans l'état } S_j \text{ en observant le symbole } y_k}{\text{nombre de fois où l'on est dans l'état } S_j}$$

Pour calculer (estimer) ces nouvelles probabilités, l'algorithme de Baum-Welch utilise deux nouvelles matrices :  $\mathbb{F}$ ,  $\mathbb{T}$ . Les coefficients de  $\mathbb{F}$ ,  $\xi_t(i,j)$ , représentent la probabilité d'être dans l'état  $S_i$  à l'instant  $t$  et de passer dans l'état  $S_j$  à l'instant  $t+1$ , d'après le modèle  $\lambda$  et la séquence d'observation O. Les coefficients de  $\mathbb{T}$ ,  $\gamma_t(i)$ , représentent la probabilité d'être dans l'état  $S_i$  à l'instant  $t$ , sachant l'observation O et le modèle  $\lambda$ .

Le calcul de ces coefficients est rapidement réalisé grâce aux deux matrices de paramètres  $\alpha$  et  $\beta$ , délivrées par les algorithmes Forward et Backward.

Calcul des coefficients  $\xi$  de la matrice  $\mathbb{F}$  :

$$\xi_t(i, j) = \frac{\alpha_t(i) \times a_{ij} \times b_j(o_{t+1}) \times \beta_{t+1}(j)}{P(O = o|\lambda)} \quad (18)$$

Calcul des coefficients  $\gamma$  de la matrice T :

$$\gamma_t(i) = \sum_{j=1}^N \xi_t(i, j) = \frac{\alpha_t(i)\beta_t(i)}{P(O/\lambda)} \quad (19)$$

Ainsi, la réestimation des paramètres (A, B et  $\Pi$ ) du modèle Markovien  $\lambda$  est telle que :

$$\pi_i = \gamma_1(i) \quad 1 \leq i \leq N \quad (20)$$

$$\bar{a}_{ij} = \frac{\sum_{t=1}^{T-1} \xi_t(i, j)}{\sum_{t=1}^{T-1} \gamma_t(i)} \quad 1 \leq i \text{ et } j \leq N \quad (21)$$

$$\bar{b}_i(k) = \frac{\sum_{t=1 \cap o_t=v_k}^T \gamma_t(i)}{\sum_{t=1}^T \gamma_t(i)} \quad 1 \leq i \leq N \quad (22)$$

L'algorithme de Baum-Welch se construit :

Faire

Appliquer les algorithmes Forward et Backward

**Pour** t=1 à T Faire

**Pour** i=1 à N Faire

**Pour** j=1 à N Faire

Calculer  $\xi_t(i, j)$

**FinPour**

Calculer  $\gamma_t(i)$

**FinPour**

**FinPour**

Réestimer  $\lambda=(A, B, \Pi)$

TantQue (il y a augmentation de  $P(O/\lambda)$ ) ou (il y a des itérations à faire)

## V. Conclusion

Nous avons défini dans ce chapitre les notions de base des modèles de Markov cachées, c'est une méthode qui a été utilisée, au début pour la prévision et puis elle a été appliquée dans le domaine de la reconnaissance et de la classification en général. Dans le chapitre suivant, nous décrivons de manière plus détaillée la partie conception de notre projet.

## CHAPITRE III. CONTRIBUTION ET CONCEPTION « ÉTUDE PREALABLE »

---

### **I. Introduction**

Cette partie est consacrée aux étapes fondamentales pour le développement de notre système de reconnaissance. Pour la conception et la réalisation de notre application, nous avons choisi de modéliser avec le formalisme UML qui offre une flexibilité marquante qui s'exprime par l'utilisation des diagrammes. La conception de notre application se base sur les diagrammes des cas d'utilisation et les diagrammes de classes.

### **II. Contribution**

La reconnaissance de visage demeure un problème complexe et non parfaitement résolu, malgré tous les travaux réalisés au cours des dernières années. Dans toute étude de la reconnaissance faciale, l'objectif ou l'objectif principal est d'atteindre un taux de reconnaissance élevé, c'est-à-dire de classifier avec précision.

Dans ce travail, nous nous sommes intéressés à l'identification faciale par l'approche Markovienne. Le système a pour but de simplifier le déroulement de l'identification qui consiste à la recherche de l'identité requête parmi l'ensemble des individus préenregistrée.

#### **II.1 Le système d'apprentissage**

Notre système d'apprentissage basé sur l'enregistrement des paramètres définissant les visages (paramètres CMC). L'utilité de CMC utilise toute information du visage, l'algorithme de Baum-Welch donne une CMC les plus fortes probabilités pour le sauvegardés.

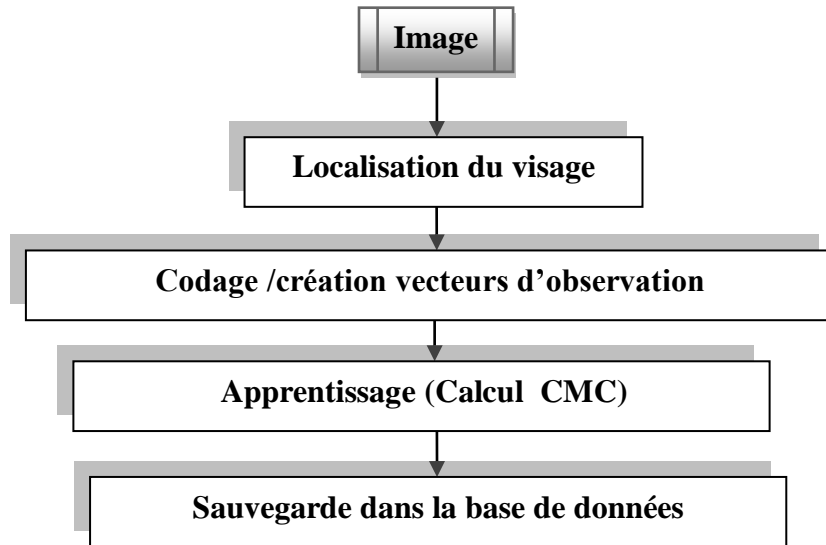


Fig.17. Les étapes de l'apprentissage

## II.2 Le système de reconnaissance

Le système de reconnaissance prend l'image d'entrée inconnue (visage), le système doit calculer les CMC par algorithme Forward/Bakward (sa probabilité d'engendrer visage à reconnaître) et bien sur après phase de codage, on obtient un score et évalue avec la base de donnée.

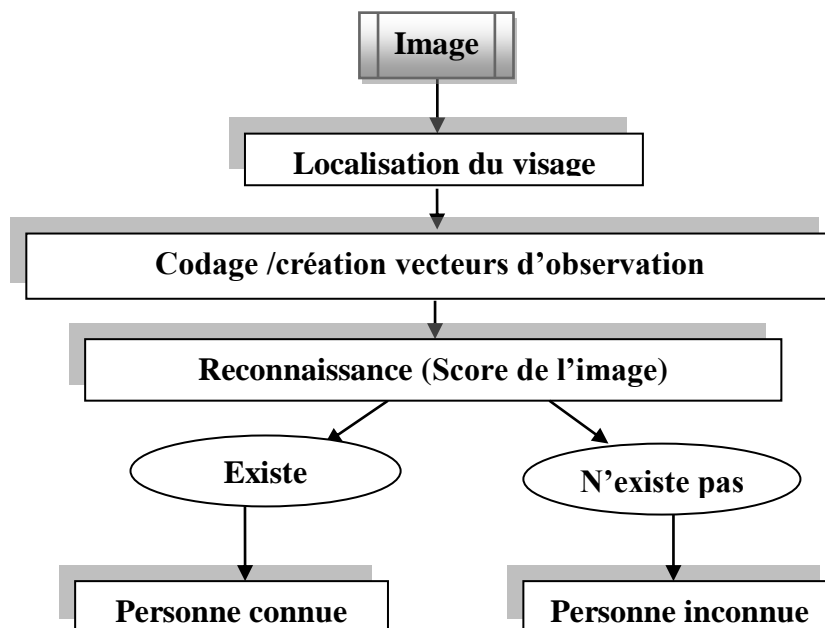


Fig.18. Les étapes de reconnaissance

### III. Conception en UML

UML est devenu un langage de conception éprouvé et bien accepté pour exprimer le système ainsi que l'architecture logicielle, c'est un langage compréhensible par les humains et par les machines. Il permet de décrire de manière très visuelle les interactions entre les différentes composantes d'un système, afin de spécifier le travail de développement attendant et de fixer des objectifs clairs

#### III.1 Diagramme Uses case

##### III.1.1 Le système d'apprentissage

Le diagramme de cas d'utilisation nous présente les principales fonctions du système, ainsi que les acteurs qui interviennent.

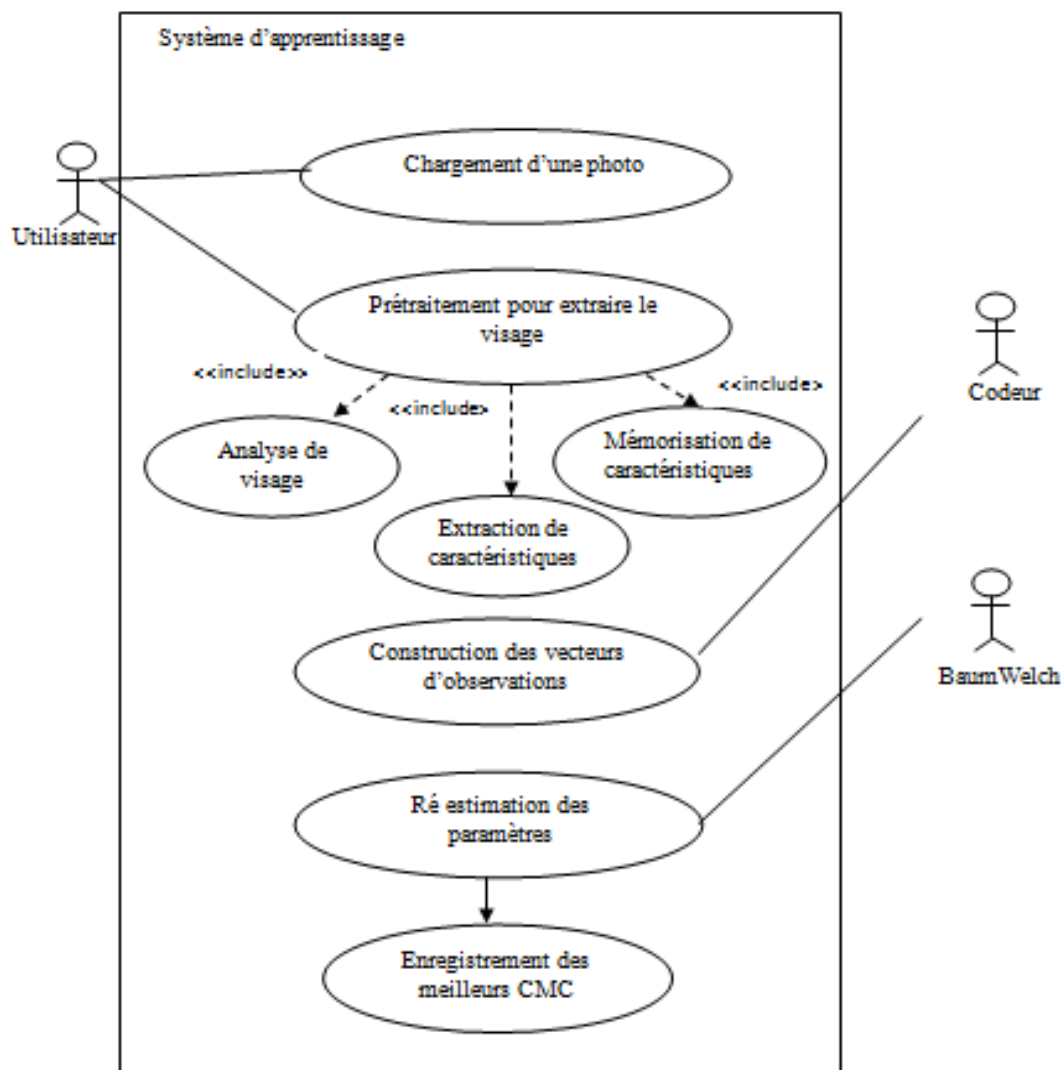


Fig.19. Diagramme de cas d'utilisation de système d'apprentissage

Les principaux acteurs de système d'apprentissage sont les utilisateurs, le codeur, BaumWelch.

L'utilisateur doit sélectionner une nouvelle photo, une fois celle-ci est sélectionnée, il peut lancer le prétraitement pour extraire le visage. Le module de codage intervient afin de construction les vecteurs d'observations. Enfin BaumWelch donne une CMC les plus fortes probabilités pour le sauvegardés.

### **III.1.2 Le système de reconnaissance**

Les principaux acteurs de système de reconnaissance sont les utilisateurs, le codeur, Forward/backward.

La reconnaissance se fera en calculant pour chaque CMC connue sa probabilité d'engendrer l'image à reconnaître. On reconnaîtra l'image dont la CMC associée a obtenu un score maximum.

Le système de reconnaissance utilise le score du modèle (donné par l'algorithme Forward/backward) pour discriminer la meilleure solution. Ce système va faire une comparaison entre les valeurs de Forward/backward et la base de données enregistré.

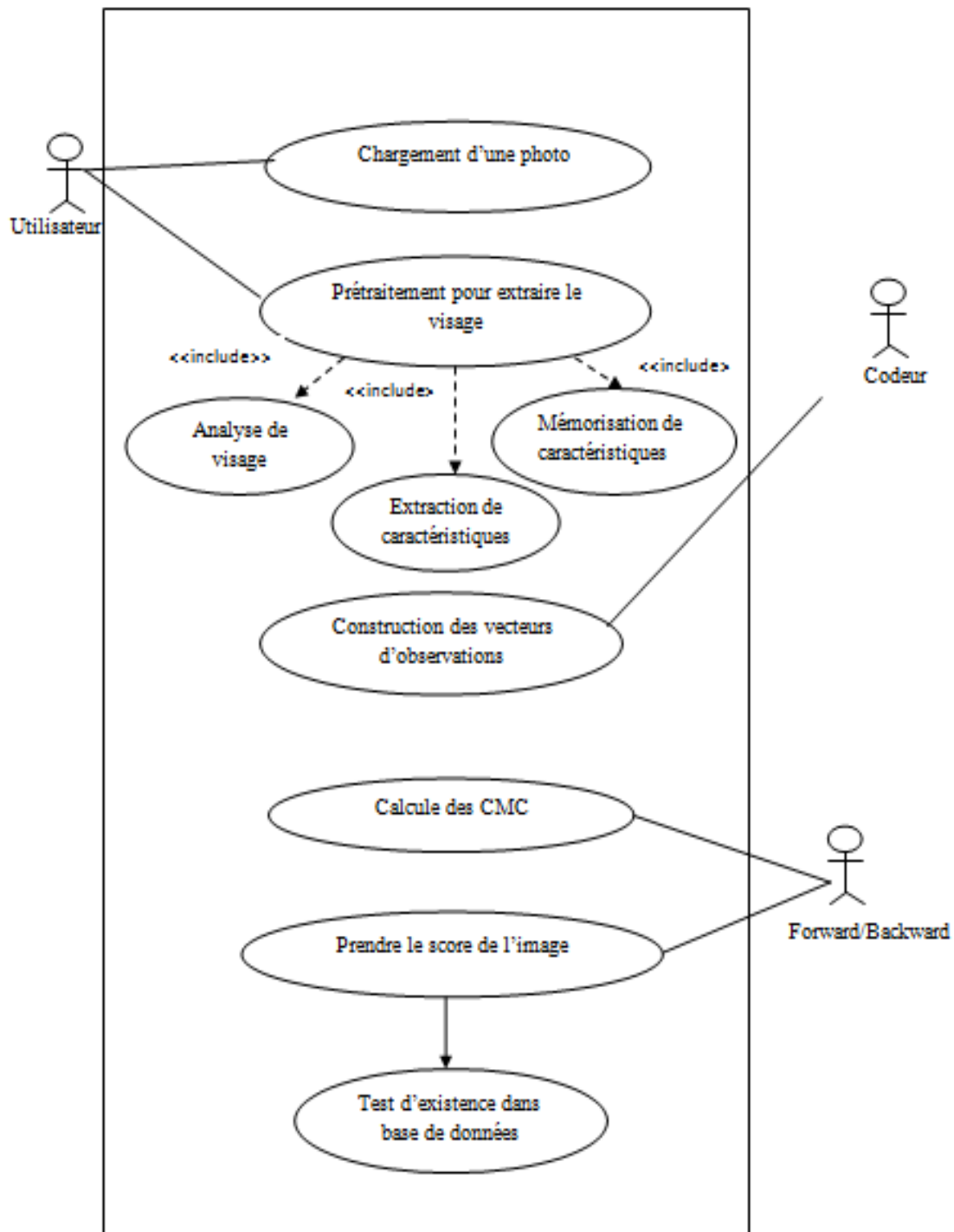


Fig.20. Diagramme de cas d'utilisation de système reconnaissance

### III.2 Diagramme de classe

Le diagramme de classe définit la structure interne d'un système, fournit une représentation abstraite des objets qui interagissent pour réaliser les cas d'utilisation.

Le diagramme de classe ci-dessous est composé de deux classes :

- La classe « Tab\_Image » contient la fonction de téléchargement de photo et la saisie de renseignement de personne.
- La classe « Tab vect\_obs » représente la classe principale du système puisqu'elle contient les fonctions de bases (fonctions de prétraitement de l'image et les fonctions de calcul de vecteur d'observation par BaumWelsh ).

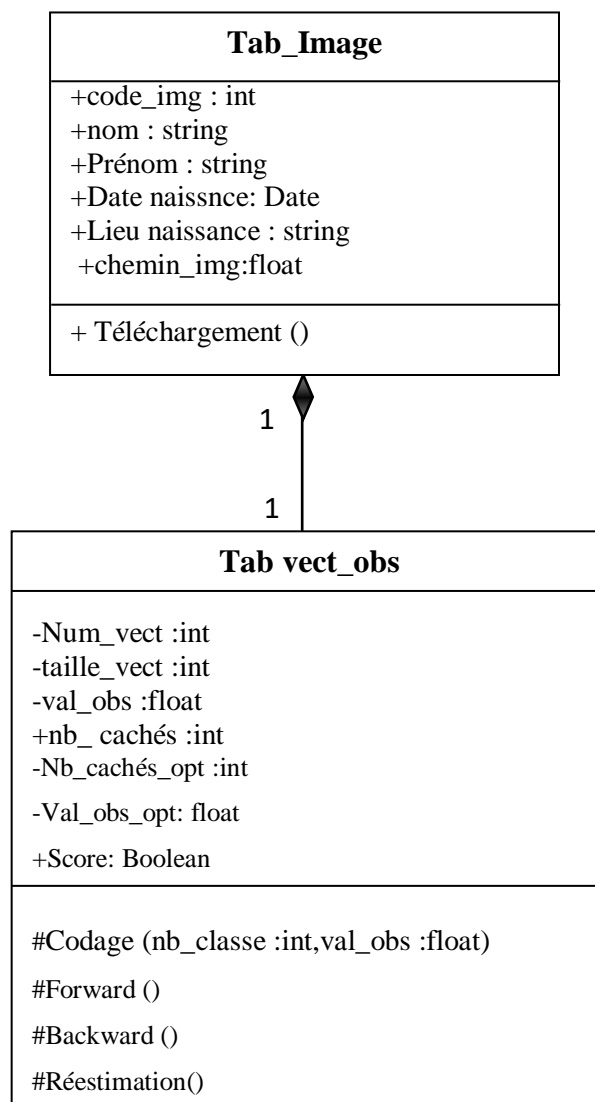


Fig.21. Diagramme de Classe

### **Conclusion**

Ce chapitre a donné une vision sur notre travail, et a donné l'aspect conceptuel de l'application de reconnaissance faciale à travers les diagrammes décrits en UML, deux diagrammes qui illustrent les différents cas d'utilisations des deux systèmes et deux diagrammes qui représentent les différentes classes métiers de nos deux systèmes.

Le chapitre qui suit fera l'objet de la mise en œuvre de notre application.

## CHAPITRE IV. IMPLEMENTATION ET DISCUSSION « ETUDE DETAILLEE »

---

### I. Introduction

Nous allons, dans cette dernière partie, mettre en application la reconnaissance faciale à base de Chaîne de Markov Cachée. Pour cela, nous commencerons par décrire notre système de reconnaissance de visage. Par la suite nous présenterons les résultats obtenus à la suite des tests effectués.

### II. Aspect matériel

Notre projet a été développé sur un micro portable:

Processeur : Intel®Core i3

Capacité Mémoire (RAM) : 4G

Capacité disque dur : 500G

Système d'exploitation : Windows 7 Professionnel.64 bit

### III. Outils de développement

Le langage que nous avons choisi c'est C++, l'un des langages de programmation les plus utilisés. Il est à la fois facile à utiliser et très efficace. La complexité du langage est inévitable lorsqu'on cherche à avoir beaucoup de fonctionnalités. En revanche, en ce qui concerne la lisibilité des programmes, tout dépend de la bonne volonté du programmeur.

➤ Les principaux avantages du C++ sont les suivants :

- Grand nombre de fonctionnalités ;
- Performances du C ;
- Facilité d'utilisation des langages objets ;
- Portabilité des fichiers sources ;
- Facilité de conversion des programmes C en C++, et, en particulier, possibilité d'utiliser toutes les fonctionnalités du langage C ;
- Contrôle d'erreurs accru.

Et nous utilisons l'enregistrement de base de données avec les tables paradox7.

#### IV. But du logiciel

Le but visé est de concevoir un système d'identification d'identité qui soit facile et peu coûteux dans l'implémentation et qui utilise une modalité biométrique particulière. L'idée principale de ce travail est d'utiliser les Chaines de Markov cachés pour l'extraction des caractéristiques de l'image de visage. Nos expériences ont été exécutées sur un ensemble des images de la base de données XM2VTS (Extended Multi Modal Verification for Teleservices and Security application).

#### V. Architecture structurelle du système de reconnaissance proposé

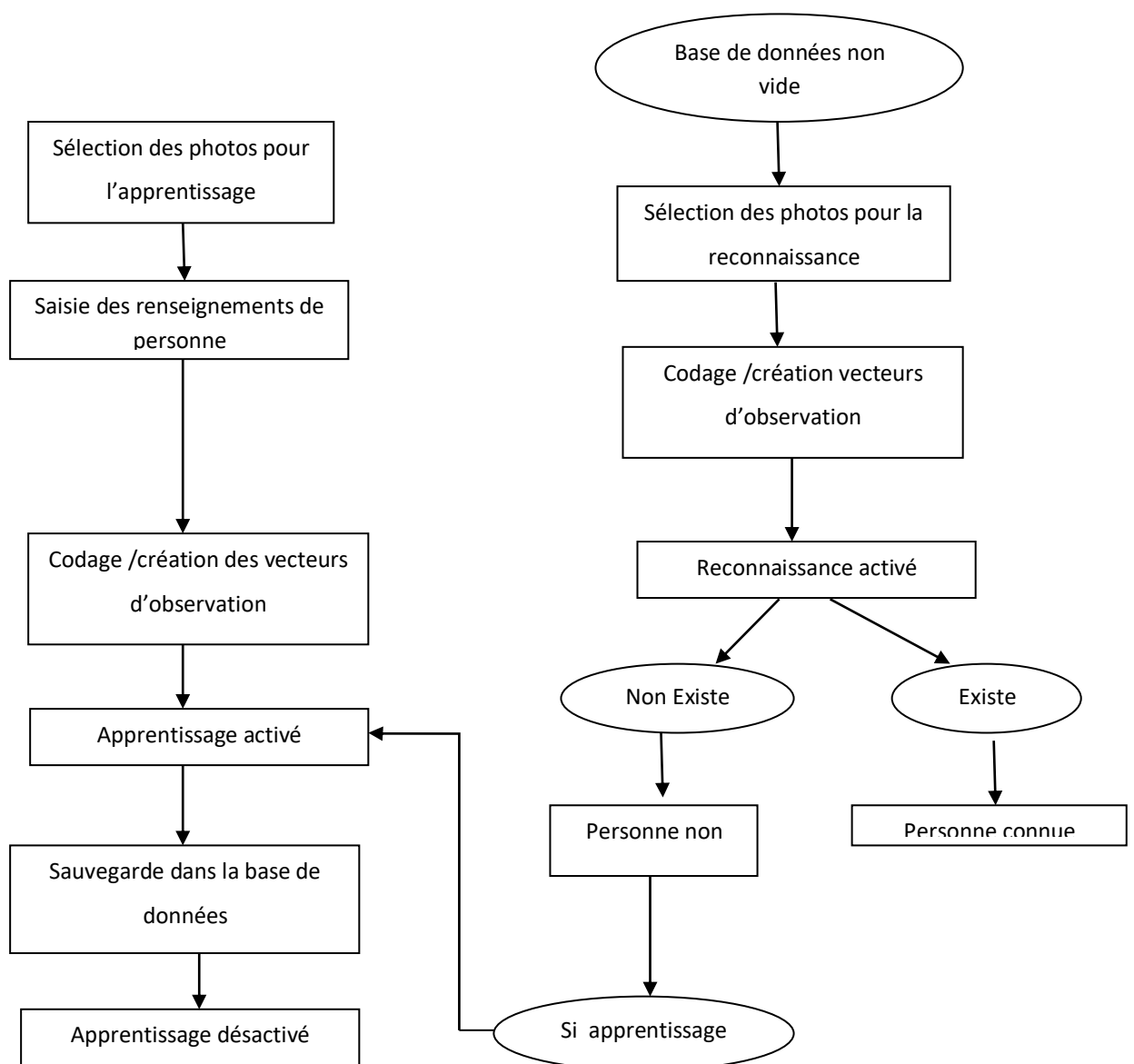


Fig.22. Architecture structurelle de notre système

## VI. Modélisation

### VI.1 Paramètres CMC

Nos paramètres sont :

- $\lambda_X = \{\pi_k, t_{ij}\}$  : paramètres a priori
  - $\pi_k$  vecteur initial :  
vectint[i]= Pixel cachés (i=Noir or Blanc) / (h\*w)-erreur;
  - $t_{ij}$  matrice de transition :  
Matrice des probabilités de transition entre Niveaux de gris des pixels (Blanc/Noir)

$\lambda_Y = \{\mu_k, \sigma_k\}$  : la loi d'attache aux données tel que  $\mu_k$  et  $\sigma_k$  la moyenne et l'écart-type respectivement

- $f_k(y_n)$  : la vraisemblance des observations  $y_n$  conditionnellement à  $X_n = \omega_k$ .

Nous allons baser sur l'algorithme de Baum-Welch pour estimer ces paramètres

#### VI.1.1 Procédure Forward ( $\alpha_n$ )

$$\alpha_n(i) = p(X_n = \omega_i, Y_1 = y_1, \dots, Y_n = y_n) \quad \forall n \in S \quad (23)$$

Cette expression peut se calculer de la manière récursive suivante :

- ✓ Initialisation:

$$\alpha_1(i) = \pi_i f_i(y_1) \quad \text{pour } 1 \leq i \leq K. \quad (24)$$

- ✓ Induction :

$$\alpha_{n+1}(i) = [\sum \alpha_n(j) t_{ji}] f_i(y_{n+1}) \quad \text{pour } 1 \leq i \leq K, 1 \leq n \leq S-1 \quad (25)$$

#### VI.1.2 Procédure Backward ( $\beta_n$ )

La probabilité backward  $\beta_n(i)$  est définie par :

$$\beta_n(i) = p(Y_{n+1} = y_{n+1}, \dots, Y_N = y_N / X_n = \omega_i) \quad \text{pour } 1 \leq i \leq K, \forall n \in S \quad (26)$$

Cette expression donne la probabilité conjointe de la séquence partielle de l'observation de l'instant n+1 jusqu'à l'instant N, sachant qu'à l'instant n, le système est dans la classe  $\omega_i$ .

Le calcul de  $\beta_n$  s'obtient à partir de  $\beta_{n+1}$  par une récurrence :

✓ Initialisation :

$$\beta_s(i)=1 \quad \text{pour } 1 \leq i \leq k \quad \forall n \in S \quad (27)$$

✓ Induction :

$$\beta_n(i) = \sum t_{ij} f_j(y_{n+1}) \beta_{n+1}(j) \quad \text{pour } 1 \leq i \leq k, 1 \leq n \leq S-1 \quad (28)$$

La procédure de calcul de la probabilité forward devient :

✓ Initialisation :

$$\text{Pour } 1 \leq i \leq K \quad \alpha_1(i) = (f_i(y_1) \pi(i)) / (\sum f_j(y_1) \pi(j)) \quad (29)$$

✓ Induction :

Pour  $1 \leq i \leq K$  et  $n=2, \dots, S$ .

$$\alpha_{n+1}(i) = f_i(y_n) [\sum \alpha_{n-1}(j) t_{ij}] \quad (30)$$

La procédure de calcul de la probabilité backward :

✓ Initialisation :

$$\text{Pour } 1 \leq i \leq K \quad \beta_n(i)=1. \quad (31)$$

✓ Induction : Pour  $1 \leq i \leq K, 1 \leq j \leq K$ , et  $n=2, \dots, S$

$$\beta_n(i) = \sum t_{ij} f_j(y_{n+1}) \beta_{n+1}(j) \quad (32)$$

## VI.2 Algorithme

### VI.2.1 Apprentissage

Input : Image à identifier

Output : Nom\_img , Cod\_img, Vect\_n, Vect\_b

Pour chaque Image<sub>i</sub> faire

- Convertit dans l'espace Ycbcr,
- Localiser le visage  $\text{tailvect}=(h*w)\text{-taierr}$ ,
- Convertit en niveau de gris,
- Créer les vecteurs d'observations  $\text{Vect}[i]$ ,
- Calculer  $\alpha_n(i)$  avec l'algorithme Forward,
- Calculer  $\beta_n(i)$  avec l'algorithme Backward,
- Appliquer l'algorithme Baum-Welsh,
- Sauvegarder les paramètres CMC.

Fin pour

### VI.2.2 Reconnaissance

Input : Image de test

Output : Nom\_img , Cod\_img, Vect\_n, Vect\_b

Pour chaque Image<sub>i</sub> faire

- Convertit dans l'espace Ycbr,
- Localiser le visage  $tailvect=(h*w)-taierr$ ,
- Convertit en niveau de gris,
- Créer les vecteurs d'observations Vect[i] ,
- Calculer  $\alpha_n(i)$  avec l'algorithme Forward,
- Calculer  $\beta_n(i)$  avec l'algorithme Backward,
- Crée un score de comparaison,
- Evaluer la correspondance entre CMC et les CMCs de la base de données,
- Prendre le meilleur score.

Fin pour

## VII. L'interface de l'application

Dans la conception de l'interface de notre application nous avons respecté un ensemble des choix ergonomiques comme la lisibilité, la compréhensibilité, etc.

C'est une interface destinée aux utilisateurs, elle est simple et permet d'illustrer les principaux processus du système de reconnaissance.

### VII.1 L'interface principale

Au lancement de notre application, une fenêtre s'affiche à l'écran, elle contient sept (07) boutons qui sont :

- « Charger une photo ».
- « Prétraitement ».
- « Extraction de caractéristiques ».
- « Mémorisation de caractéristiques ».
- « Reconnaissance ».
- « Annuler ».
- « Consulter la base de données ».

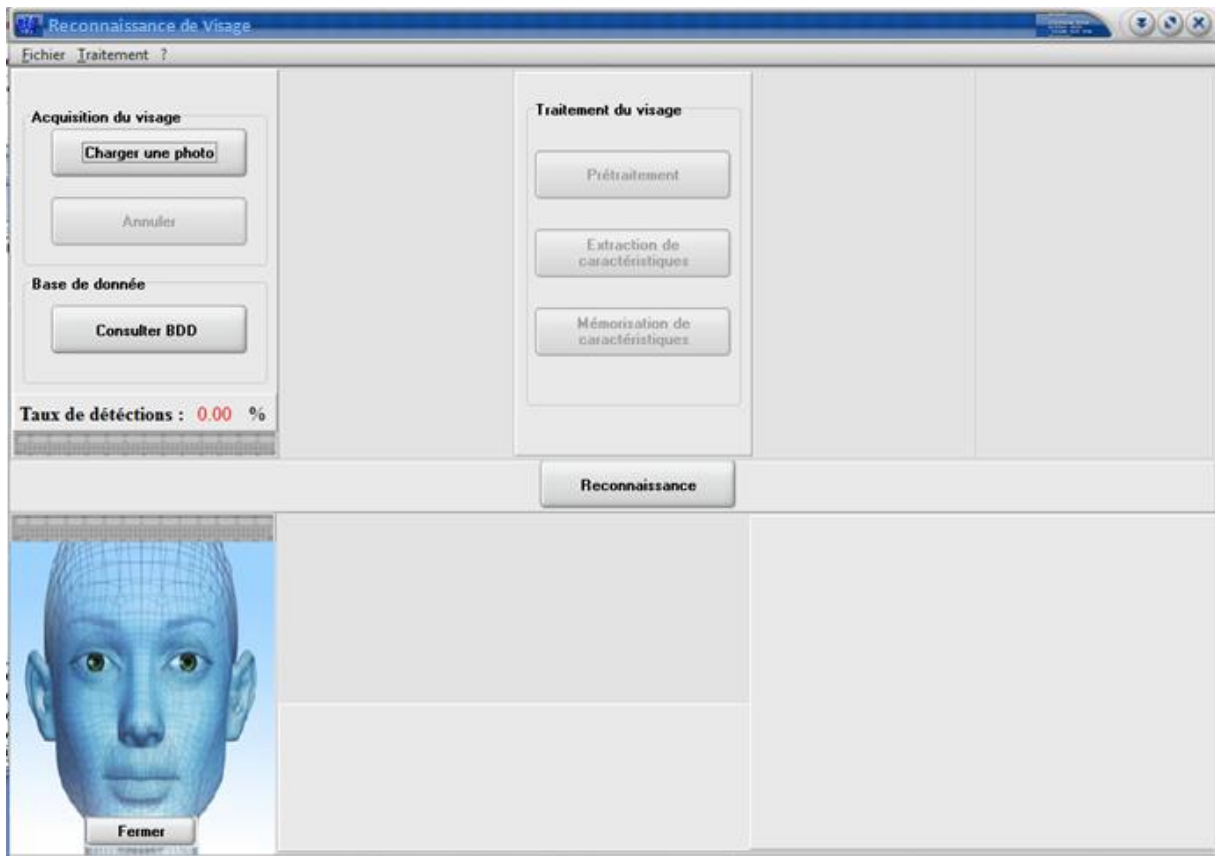


Fig.23. Interface de l'application

La fenêtre d'interface illustrée par la figure 24 contient également un menu qui contenant les différentes fonctions de notre application.

### VII.2 L'interface de la base des données

L'interface de la base des données contiennent une liste des noms, des prénoms, des adresses et des aperçus des personnes connues, comme le présente la figure 25

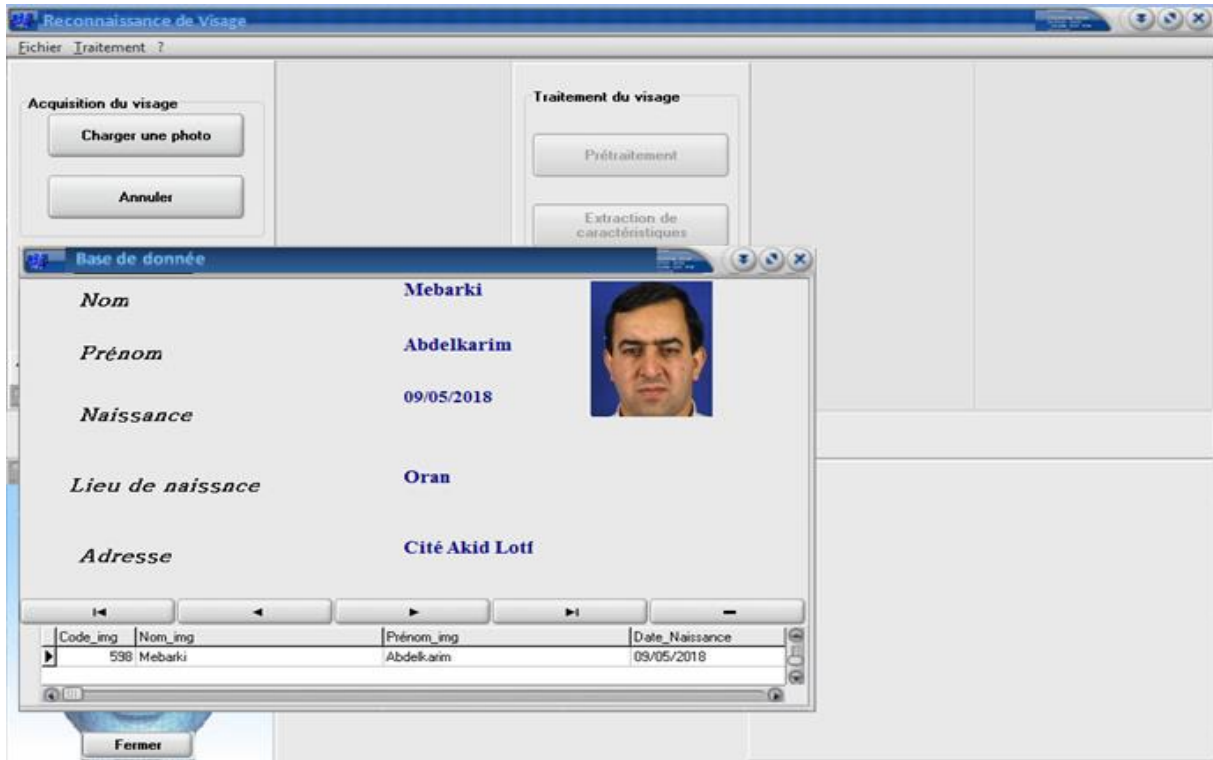


Fig.24. L'interface de la base des données

### VII.3 L'interface de base de caractéristiques

Cette interface représente notre base de caractéristiques qui contient les différentes caractéristiques des visages (les vecteurs d'observations).

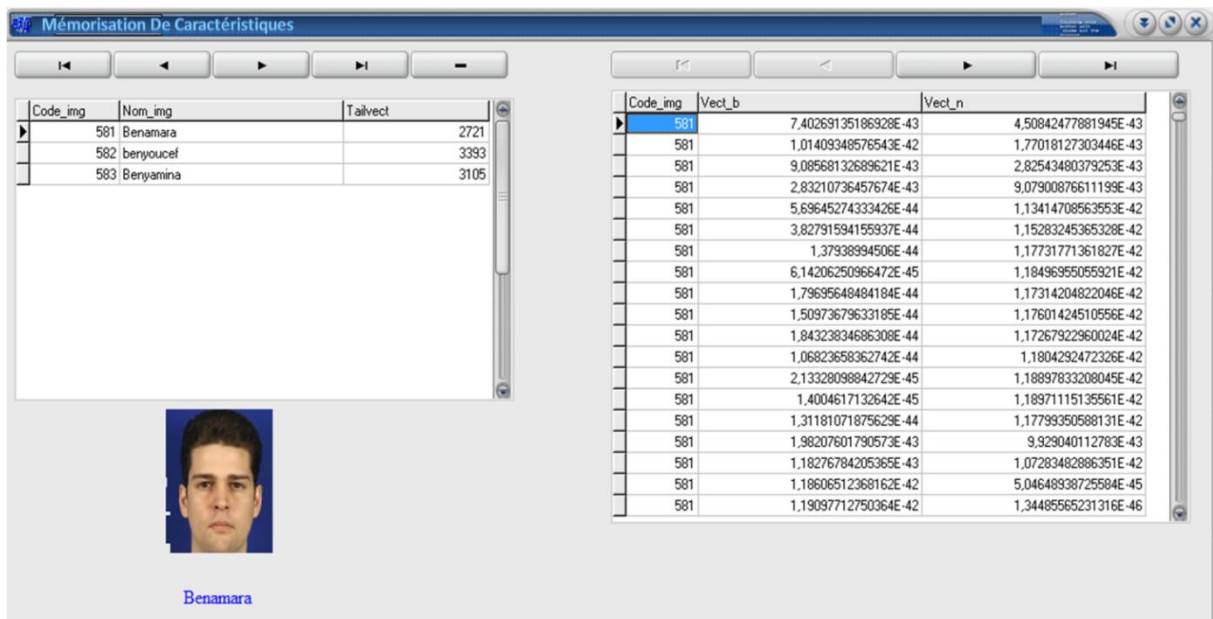


Fig.25. base de caractéristiques

## VIII. Déroulement de l'application

### VIII.1 L'apprentissage

#### VIII.1.1 Chargement de photo

On click sur le bouton « Charger une photo » ensuite on sélectionne la photo à traiter.

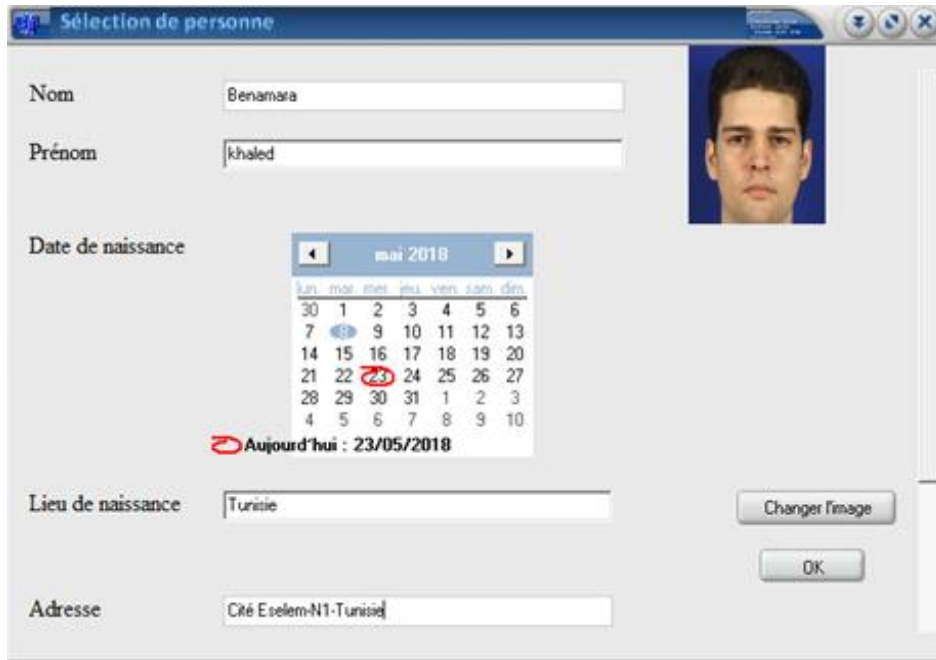


Fig.26. Charger la photo à rechercher

Quand le chargement est terminé on reçoit un message box "Chargement avec succès"

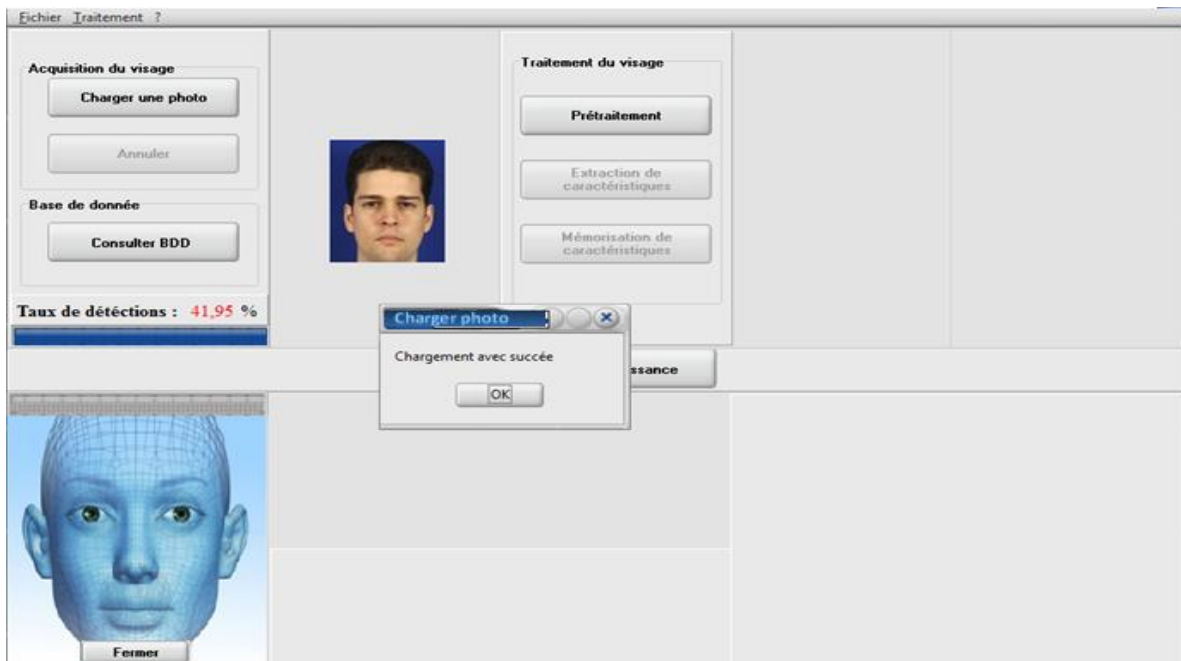


Fig.27. Message fin chargement

### VIII.1.2 Prétraitement de photo

C'est une étape de prétraitement pour transformer dans l'espace couleur YCbCr

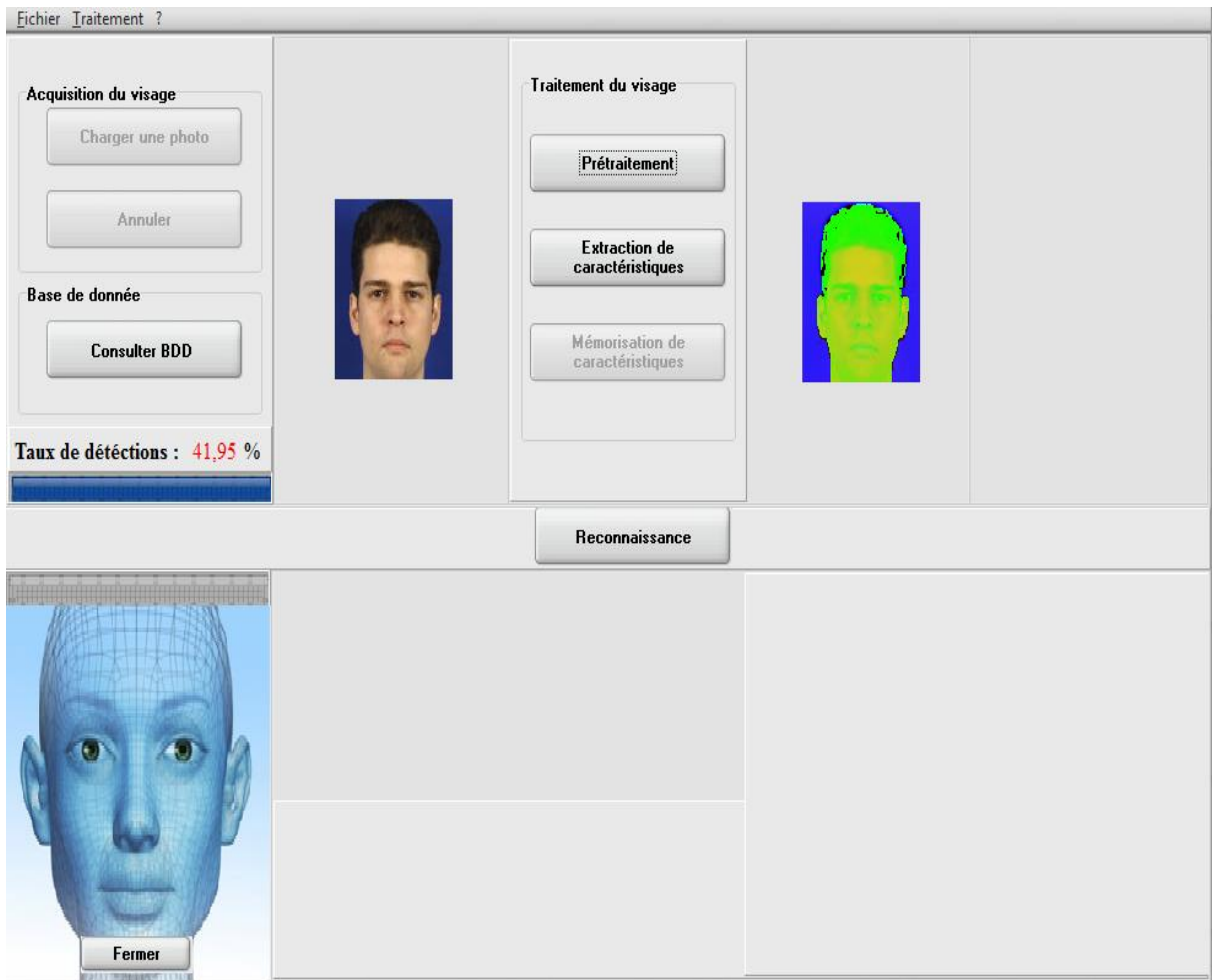


Fig.28. Prétraitement de photo

### VIII.1.3 Extraction de caractéristiques

C'est l'étape d'extraction automatique de points caractéristiques sur le visage. Le visage détecté va transformé ou codé au niveau de gris automatiquement.

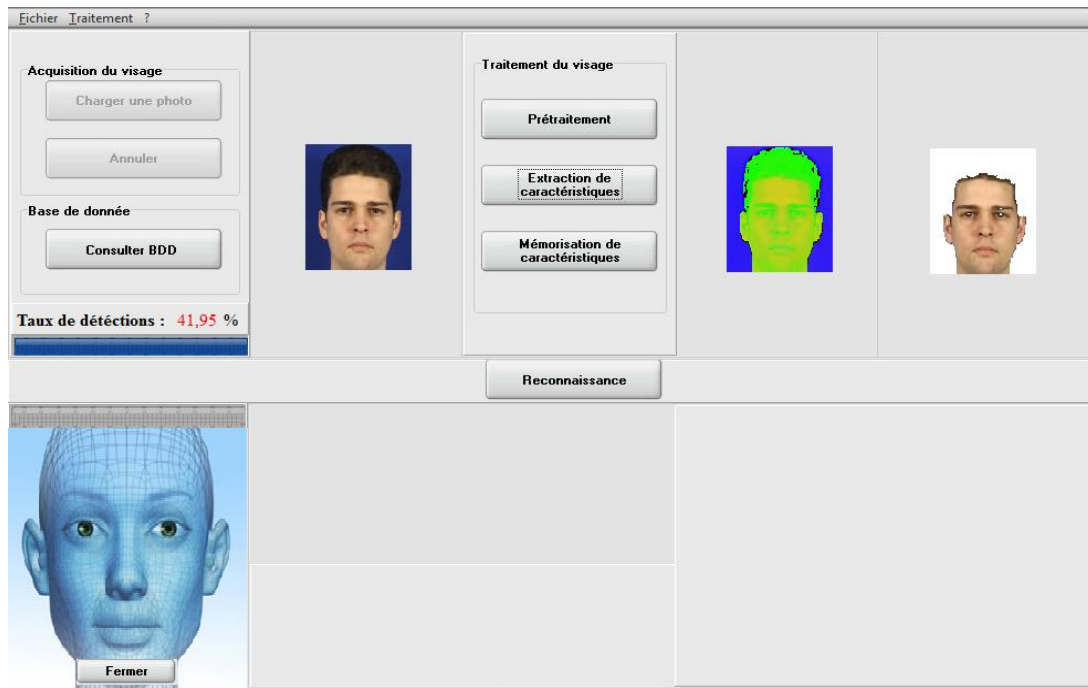


Fig.29. Extraction de caractéristiques

#### VIII.1.4 Mémorisation de caractéristiques

Dans ce cas le programme calcule les vecteurs d'observations pour chaque image et sauvegarde les résultats d'apprentissage dans deux tableaux « tab\_img, tab\_vect »

Tab\_img (Code\_img, Nom\_img, Prénom\_img, Date\_naissance, Lieu\_naissance, Adresse chemin\_img) ;

Tab\_vect (code\_vect, vect1, vect2, code\_img\*) ;

Il ya une relation père fils entre les deux tableaux.

a) Pour tab\_img :

- Le code\_img : ils incrémentent automatique.
- Nom\_img , Prénom\_img, Date\_naissance, Lieu\_naissance, Adresse: Donne par l'utilisateur.
- Chemin\_img : Prendre le répertoire de l'image.

b) Pour tab\_vect :

- Le code\_vect : ils incrémentent automatique.
- Vect1, vect2 : sont les valeurs après apprentissage.

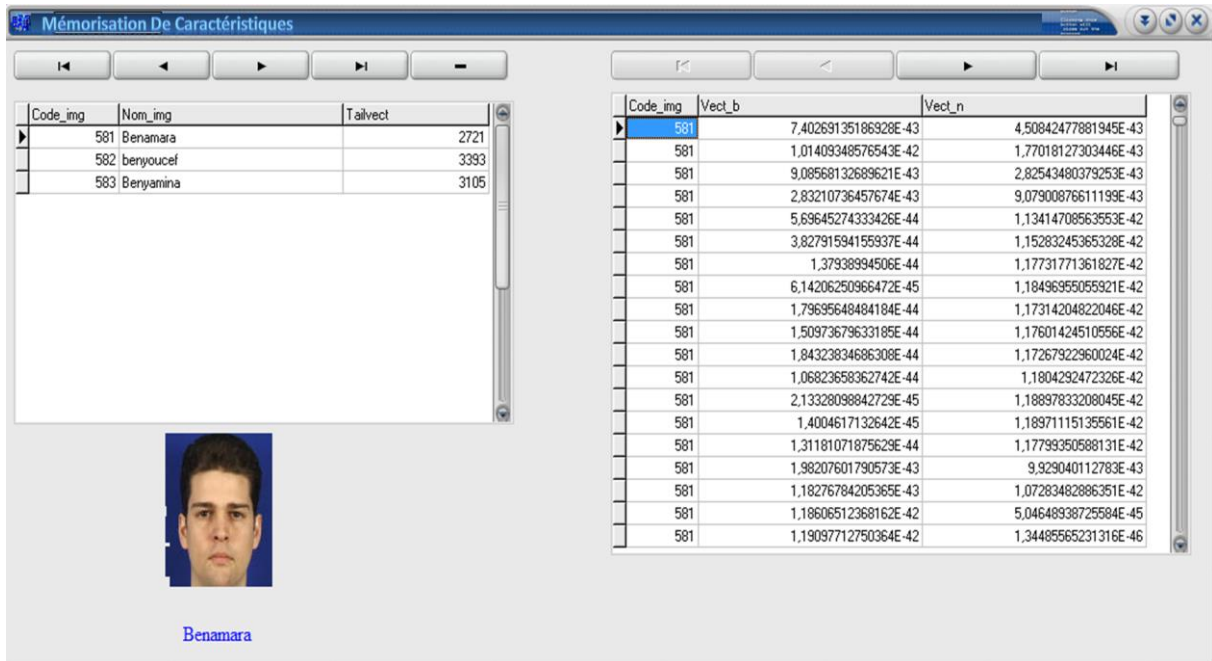


Fig.30. Mémorisation de caractéristiques

## VIII.2 Reconnaissance

Le système de reconnaissance se déroule en cinq étapes :

1. Importer une image de la base de test
2. prétraitement,
3. extraction de caractéristiques faciales,
4. Importer une image de la base de test,
5. Comparer cette image avec ceux stocké dans la base de données.

La reconnaissance donne deux résultats :

- Personne connue : le système affiche les renseignements de cette personne.
- Personne inconnue : le système affiche que cette personne n'est pas connue.

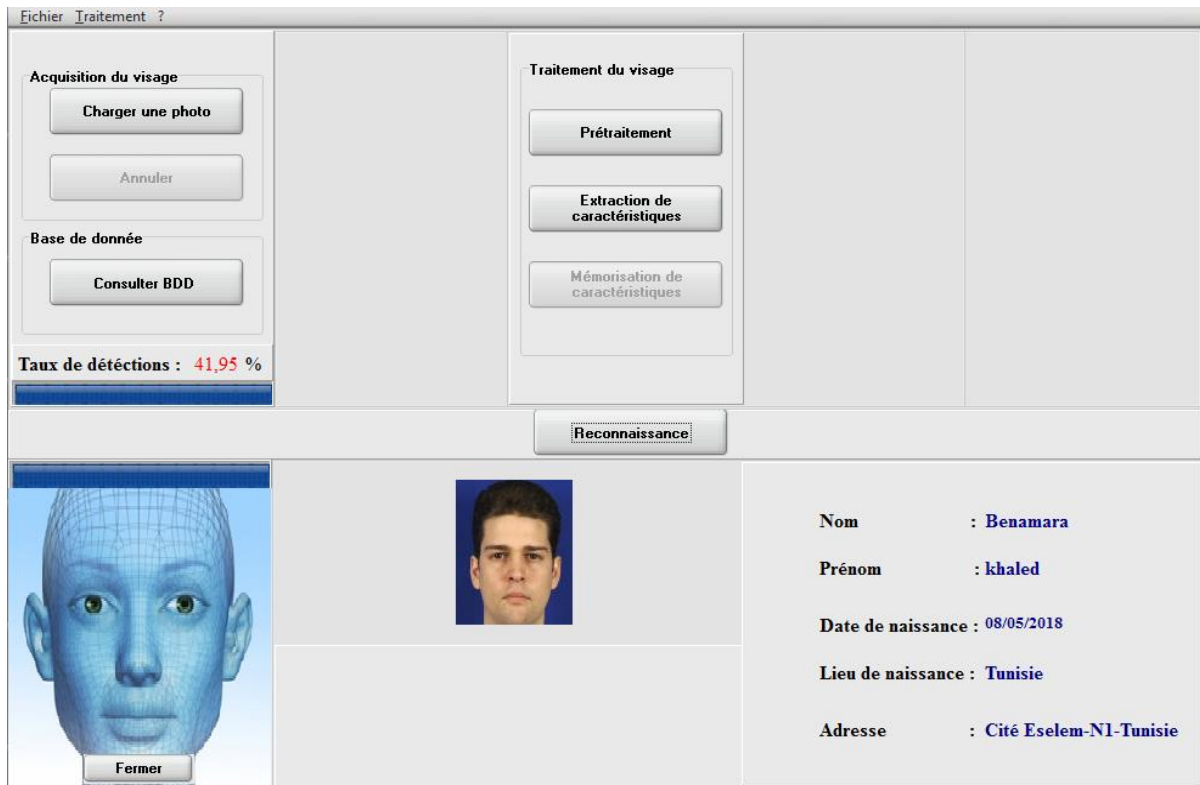


Fig.31.personne connue

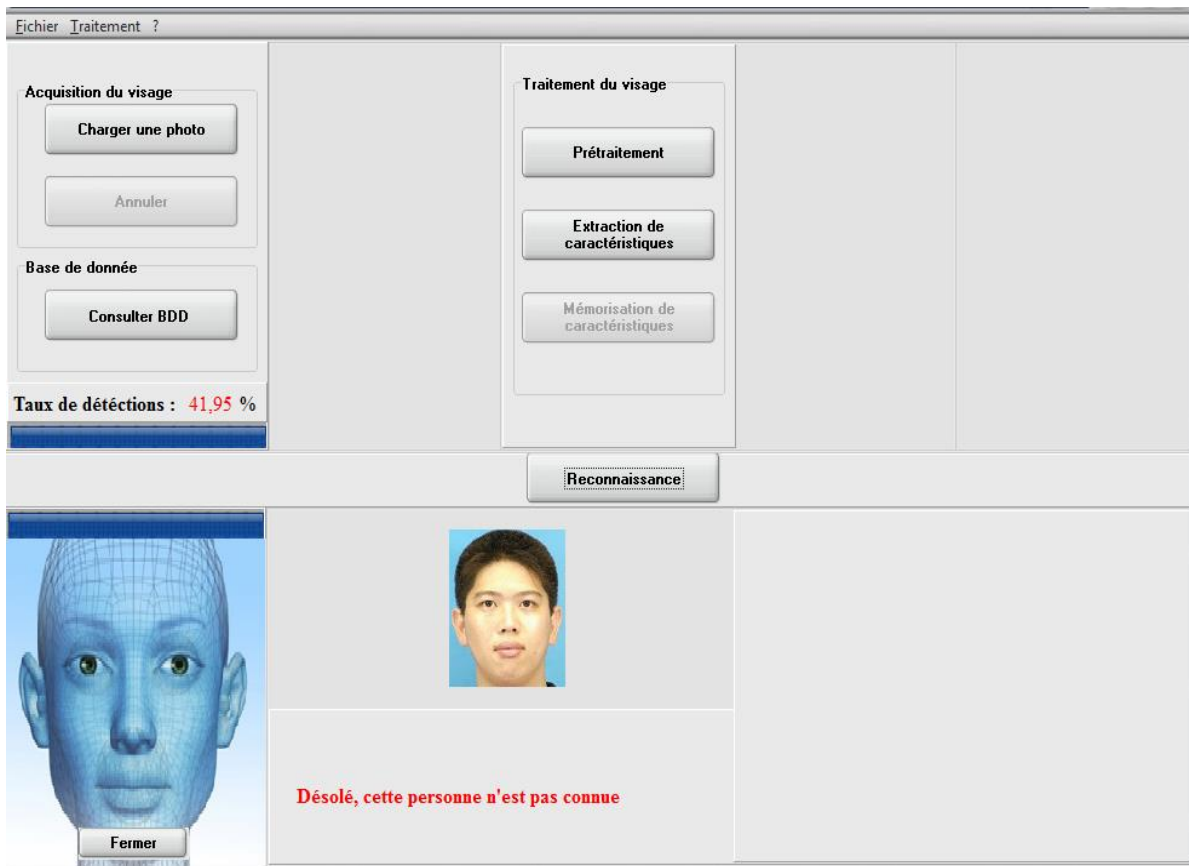


Fig.32.personne inconnue

### IX. Résultats et Expérimentations

Nous avons expérimentée notre travail sur différents types d'image pour tester sa capacité à identifier les personnes.

La base de données de visages que nous avons travaillée sur laquelle est la base de données prolongée de M2VTS, celle qui a été enregistrée pendant le projet européen M2VTS (Multi Modal Vérification for Teleservices and Security applications).Ce projet traite le contrôle d'accès par l'utilisation de l'identification.



Fig.33.Exemples des images de la base de données XM2VTS

#### IX.1 Expériences

Un ensemble d'expériences a été réalisé selon les conditions suivantes :

- Orientation et pose.
- Conditions d'illumination et qualité de l'image.
- Variabilité intrinsèque des visages humains (couleur, taille, forme).
- Présence caractéristiques particulières (cheveux, moustache, barbe, lunettes...).
- Expressions faciales modifiant la géométrie du visage.

Le tableau 2 résume quelques résultats obtenus via les tests effectués sur des images de personnes avec différentes postures.

IX.1 .1 Orientation et pose

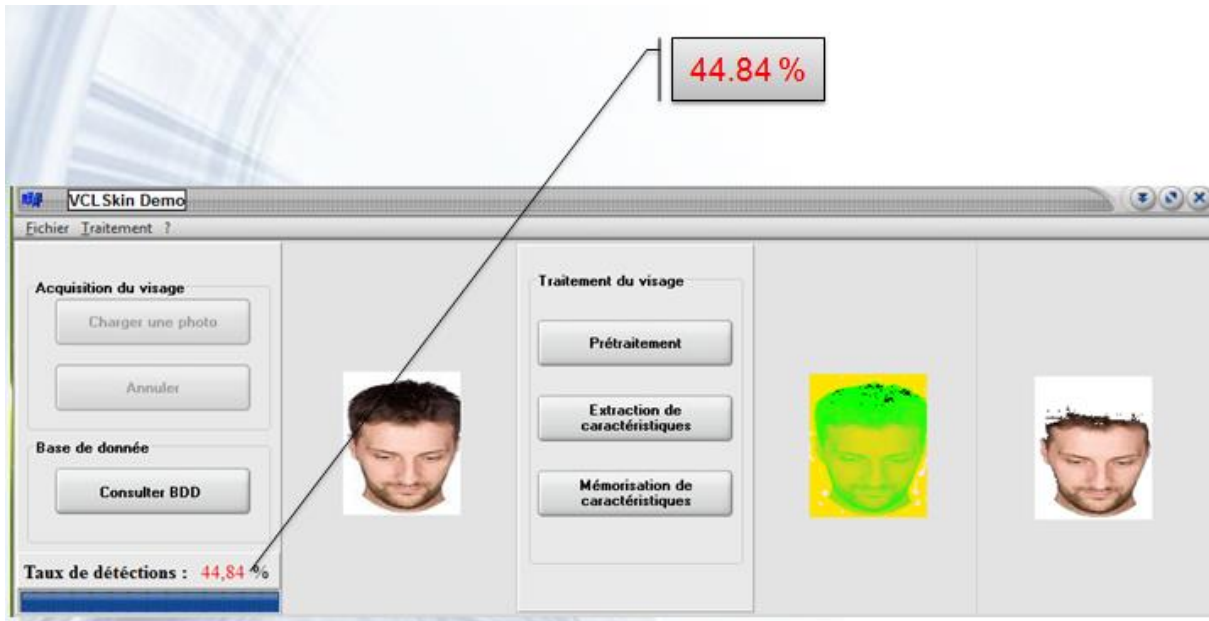


Fig.34. Orientation et pose

IX.1 .2 Conditions d'illumination



Fig.35. Conditions d'illumination

IX.1 .3 Qualité de l'image



Fig.36. Qualité de l'image

IX.1 .4 Expressions faciales



Fig.37. Expressions faciales

IX.1 .5 Présence caractéristiques particulières (une barbe)



Fig.38. Présence caractéristique particulier

Nom d'image	Image original	Après transformation	Après localisation
Image 1			
Image 2			
Image 3			








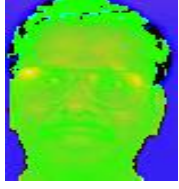



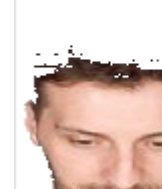

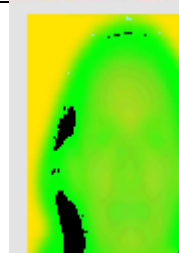

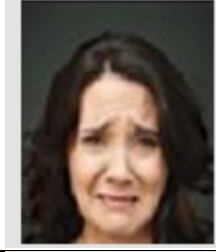

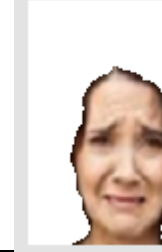

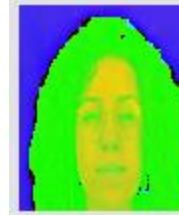
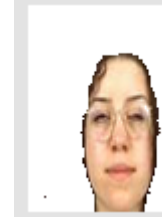
Image 4			
Image 5			
Image 6			
Image 7			
Image 8			
Image 9			
Image 10			



Tableau 2. Résultats des tests

Les résultats montrent que la variation de l'éclairage et le changement de la position du visage influence le processus de détection. En position fixe et éclairée, les taux de détection sont effectivement élevés.

### X. Conclusion

Dans ce chapitre, nous avons expliqué en détails la plateforme logicielle qu'on a développée en utilisant le langage C++. Nous avons utilisé quelques images pour tester les performances de cette implémentation et nous avons remarqué que les résultats obtenus présentaient certaines limites en particulier la position et l'intensité de l'éclairage ce qui rend le système quasi inutilisable.

À travers les tests obtenus on peut dire que la performance du système repose sur un critère, qui est le bon choix des images d'apprentissages.

### Conclusion Générale et perspectives

La reconnaissance des visages est une application parmi d'autre de la biométrie. Elle suscite de plus en plus l'intérêt de la communauté scientifique, car elle présente plusieurs challenges et verrous technologiques.

Au cours de ce travail, nous avons mis en œuvre une approche markovienne d'identification du visage. Un état de l'art des différentes approches récentes dans le domaine est établi. Une approche conceptuelle par l'UML a été abordée pour une grande partie de notre travail. Nous avons utilisé aussi Borland C++ Builder et le gestionnaire de bases de données Paradox pour implémenter notre application. Les tests ont montré que le taux de détection diminue lorsque l'orientation du visage ou la taille de l'image du visage changent. Parmi les perspectives ouvertes à ce travail, l'hybridation entre l'approche Markovienne et d'autres approches et fiabiliser le système en diminuant la sensibilité aux conditions d'éclairage, orientation et expressions faciales. Notre étude serait beaucoup plus pondérante si le système était en temps réel.

Le projet de fin d'étude a fait l'objet d'une expérience intéressante, qui nous a permis d'améliorer nos connaissances en conception et en programmation.

## BIBLIOGRAPHIE

- [Web1]** Article." reconnaissance faciale" ,Wiki,Last update.12/02/2017.  
[Online].Available: <https://labiometrie.wordpress.com/2017/02/12/reconnaissance-faciale/>  
[Accessed] :2018
- [Ani10]** Anis CHAARI, Sylvie LELANDAIS, Mohamed SAAIDIA, Vincent VIGNERON, Mohamed BEN AHMED « Comparaison d'approches pour la détection et l'identification de visages 2D », 2010.
- [Bel97]** P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman. Eigenfaces vs. Fisherfaces" Recognition using class specific linear projection". IEEE Trans. PAMI,1997.
- [Wih08]** Wihoho « Implement face recognition using PCA, LDA and LPP », github, Last Update.26/03/2018.  
[Online].Available: <https://github.com/wihoho/FaceRecognition>. [Accessed] : 2018
- [Rob00]** Robert Bergevin " Vision numérique "Université Laval, Automne 2000.
- [Guo00]** G. Guo, S.Z. Li, K. Chan, Face Recognition by Support Vector Machines, Proc. of the IEEE International Conference on Automatic Face and Gesture Recognition, 26-30March 2000, Grenoble, France, pp. 196-201.
- [Tom 97]** Tom Mitchell,home work 3 , "neural net and face image"CMU ,October1997.
- [Man13]** L. Mansoura "Identification des Visages Humains par Réseaux de Neurones"université de batna faculté de technologie 2013.
- [Min02]** Ming-Hsuan Yang, David J. Kriegman et Narendra Ahuja "Detecting faces in images : A survey". Dans IEEE Transactions on Pattern Analysis and achineIntelligence, volume 24(1), pages 34–58, 2002.
- [Ahm15]** G. Ahmed "Amélioration de la reconnaissance par le visage" Laboratoire Modélisation et Optimisation des Systèmes Industriels 2015.
- [Ale03]** Alexandre Lemieux,' système d'identification de personnes par Vision numérique' 2003.
- [Tho04]** Thomas EA, F. ROSSANT, F. AMIEL, A.AMARA N. MORIZET "Revue des algorithmes PCA, LDA et EBGm utilises en reconnaissance 2D du visage pour la biométrie"2004.
- [Shr14]** Shravan Halankar " face recognition technolog" ,Slideshare,Last update.03/12/2014.  
[Online].Available:<https://www.slideshare.net/shravanhalankar/face-recognition-technology-42335015> .  
[Accessed] :2018
- [Gar99]** Garcia C. , Tziritas G."Face Detection Using Quantized Skin Color Regions Merging and Wavelet Packet Analysis", IEEE Transactions on Multimedia, 1(3), September 1999, p.264-277.
- [Qia17]** QianLiua, ChaoWanga, Xiao-yuan Jingc "Dual multi-kernel discriminant analysis for color face recognition", Sciencedirect., Last Update. 28 /03/2017.  
[Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0030402617303704> [Accessed] :2018
- [Min02]** Ming-Hsuan Yang, David J. Kriegman et Narendra Ahuja "Detecting faces in images : A survey". Dans IEEE Transactions on Pattern Analysis and achineIntelligence, volume 24(1), pages 34–58, 2002.
- [Ank11]** Ankit Gupta "face-recognition-tech1" , slideshare,Last Update. 14 /09/2011.  
[Online].Available: <https://www.slideshare.net/ankitvg/face-recognition-tech1>  
[Accessed] :2018
- [Fer94]** Ferdinando Samaria "Face Recognition Using Hidden Markov Models". Thèse de doctorat, Engineering Department, Cambridge University, Trumpington Street, Cambridge CB2 1PZ, UK, October 1994.
- [Sam&You94]** F. Samaria and S. Young, "HMM based arhitecture for face identification, " Image and Computer Vision, vol. 12, pp. 537-583, October 1994.
- [Eic99]** Eickeler,Muller and Rigoll 1999 "high quality face recognition in JPEG compressed image" IEEE int.Conference on Image Processing Kobe.Japan 1999.
- [Oth00]** H.Othman, T. Aboulnasr, 2000 Hybrid hidden markov model for face recognition. 4th IEEE Southwest Symposium on Image Analysis and In- terpretation, 3440 , April 2000.

- [Yuj07]** Yujian, "An analytic solution for estimating two-dimensional hidden Markov models" College of Computer Science and Technology, Beijing University of Technology, Beijing 100022, China 2007.
- [Pet11]** Peter Corcoran, Claudia "Hidden Markov Models in Automatic Face Recognition" Iancu 2011 College of Engineering & Informatics National University of Ireland Galway Ireland.
- [Mar17]** Mariusz Kubanek, Janusz "The use of hidden Markov models to verify the identity based on facial asymmetry 2017" Bobulski Email author View ORCID ID profile EURASIP Journal on Image and Video Processing 2017.
- [Tim14]** Tim Chuk; Antoni B. Chan; Janet H. Hsiao "Understanding eye movements in face recognition using hidden Markov models". Journal of Vision September 2014, Vol. 14, 8. doi:10.1167/14.11.8.
- [Web2]** Article. "Eigenface", Wiki, Last update. 24/08/2016.  
[Online]. Available: <https://fr.wikipedia.org/wiki/Eigenface>.  
[Accessed] : 2018
- [Man12]** W. Manoirl & C. Kimberley Article. "method eigenface", wordpress, Last update. 10/05/2012.  
[Online]. Available: <https://reconnaissancefaciale.wordpress.com/methode-eigenface>.  
[Accessed] : 2018
- [Nic06]** Nicolas MORIZET, Thomas EA, Florence ROSSANT, Frédéric AMIEL et Amara AMARA. "Revue des algorithmes PCA, LDA et EBGm utilisés en reconnaissance 2D du visage pour la biométrie" P1-11. Institut Supérieur d'Electronique de Paris (ISEP), département d'Electronique, 2006.
- [Turk91]** M. Turk, A. Pentland, "Eigenfaces for Recognition, Journal of Cognitive Neuroscience", Vol. 3, No. 1, 1991, pp. 71-86
- [Hed15]** M. Hedouche Mémoire de Fin d'Etudes En vue de l'obtention du diplôme MASTER sous le thème " RECONNAISSANCE DE VISAGE 3D BASÉE SUR LA LOCALISATION DU NEZ " 2015.
- [Phi15]** Philpetitpa Le traitement d'images " OPENCV Reconnaissance de visages par la méthode des eigenfaces " 24/12/2015.
- [Phi06]** Philippe A. Martin, « INTRODUCTION AUX PROCESSUS STOCHASTIQUES EN PHYSIQUE », 4ème Séminaire Transalpin De Physique : Nonequilibrium Statistical Mechanics Champex 5-11 Mars 2006.
- [Bou03]** Bouhlel Nizar, « Segmentation Bayésienne Par Approche Markovienne », Université René Descartes, UFR De Mathématiques Et Informatique, MAP5. Université Tunis El Manar, Ecole Nationale D Ingénieurs De Tunis, 2003, U2S.
- [Sté12]** Stéphane DERRODE, « Introduction Aux Modèles Markoviens Pour Le Signal Et l'Image », 2012.
- [Ouh12]** Ouahab Abdelouahab « Chaînes de Markov Cachées et Application à l'image satellitaire: classification » 2012.
- [Sté08]** Stéphanie BRICQ, « Segmentation D'images IRM Anatomiques Par Inférence Bayésienne Multimodale Et Détection De Lésions », Présentée Pour Obtenir Le Grade De Docteur De l'Université Louis Pasteur – Strasbourg, 2008.
- [Pie06]** W. Pieczynski, « Chaînes De Markov Triplets Et Segmentation Non Supervisée Des Signaux », 2006.
- [Moh99]** Mohamed SLIMANE, Thierry BROUARD, Gilles VENTURINI et Jean-Pierre ASSELIN DE BEAUVILLE « Apprentissage non-supervisé d'images par hybridation génétique d'une chaîne de Markov cachée » ; par Laboratoire d'Informatique- EA2 101, Ecole d'Ingénieurs en Informatique pour l'Industrie, Université de Tours, 64, Avenue Jean Portalis, 37200 Tours France 1999.
- [Bel15]** Belbedj Farid « Les modèles de Markov Cachés et leur application dans un processus industriel Application à un système de pompage de ciment » 2015.
- [Deq05]** Dequier JEROME, 'Chaînes de Markov et applications', Centre D'enseignement De Grenoble, 2005.
- [You16]** Yousri Kessentini Modèles de Markov multi-flux pour la reconnaissance de l'écriture manuscrite multi-scripts HAL Id: tel-01415112 [https](https://hal.archives-ouvertes.fr/tel-01415112), 2016.

**[Flo07]** Florian AGEN, Julien MICHOT Projet de Mathématiques Chaines de Markov cachées Algorithme de Baum-Welch 2007.