الجمهورية الجزائرية الديمقراطية الشعبية

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTRE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE MOHAMED KHEIDER DE BISKRA

FACULTE DES SCIENCES DE L'INGENIEUR

DEPARTEMENT D'ELECTRONIQUE

Filière : Electronique

**Option : Communication**

## Mémoire de Fin d'Etude
## En vue de l'obtention du diplôme

Ingénieur de l'Electronique

***Thème***

# Détections des Visages par les réseaux de neurones

Présenté par :

- **BENZAF Naim**
- **LABED Hanene**

Proposé et Dirigé par :

**Mme BENATIA Belahcen Mebarka**

**Promotion : 2009/2010**

# Introduction generale

Aujourd'hui, l'informatique est présente dans quasiment tous les domaines. Elle est au coeur des appareils d'usage domestique tels que les micro-ondes, machine à laver, téléviseurs etc… comme elle est présente en force dans les moyens de transport tels que la voiture, l'avion, les trains … elle est également présente au centre de toutes les hyper structures modernes tels que les complexes industriels, les centrales nucléaires et l'exploration spatiale. C'est dire que l'informatique est omniprésente dans toutes les activités. Ceci nous conduit à relever l'importance de l'outil informatique pour la télésurveillance qui est à la base des systèmes de contrôle, de régulation et parfois de décision.

Notre sujet concerne donc l'exploitation d'informations capturées par des appareils adéquats afin de détecter des objets précis en l'occurrence des visages. Notre étude a des applications et des implications financières et technologiques importantes. Nous allons aborder un domaine très large qui est directement liés à la détection des visages.

La détection du visage est la première phase du système biométrique basé sur la reconnaissance du visage. Le but est de vérifier la présence d'un ou plusieurs visages sur une image..Une mauvaise localisation et/ou normalisation du visage cause une chute drastique du taux de reconnaissance et rend le système d'authentification moins performant.

Pour avoir un système d'authentification plus performant, il faut que le module de prétraitement soit le plus précis possible afin d'obtenir les meilleures conditions initiales.
L'étape de prétraitement joue le rôle d'une réduction des données ainsi qu'une atténuation des effets d'une différence de conditions lors des prises de vues.

Dans la littérature, plusieurs travaux portant sur la détection de visage et de mouvement sont souvent utilisés dans des applications de codage, de poursuite, d'indexation, de surveillance, etc…

Parmi les méthodes existantes, on cite :

- L'approche la plus répandue est sans doute, celle utilisant **un réseau de neurones** pour classifier les pixels de l'image, en tant que visage ou non-visage.

-L'approche qui consiste à déterminer des **points d'intérêts** ( **maxima locaux** d'un filtre gaussien aux dérivées secondes), en partant de ces derniers, on réalise une détection de contours qui seront examinés pour être groupés en régions. Le groupement est basé sur leur proximité et leur similarité en orientation et épaisseur

-Des études ont montré que ***la variabilité de la couleur de la peau*** tenait plus de la différence d'intensité plutôt que de la chromaticité. On peut utiliser plusieurs types d'espaces de couleurs (RGB, RGB normalisé, HSV, YcrCb, YIQ, YES, CIE XYZ, CIE LUV).

-D'autres études ont toutefois montré que la combinaison **d'analyse de formes**, de **segmentation de couleurs** et d'**informations de mouvement.**

L'approche la plus répandue est sans doute, celle utilisant **un réseau de neurones** pour classifier les pixels de l'image, en tant que visage ou non-visage.
La base de données XM2VTS est comparer les différentes méthodes de vérification d'identité.

Notre travail se présente sous forme de cinq chapitres ;

Dans le **chapitre I,** nous présentons la détection de visages ses intérêts et ses difficultés et les différentes techniques sont donnés. Il est possible de distinguer quatre approches théoriques qui sont détaillées dans ce chapitre.

Dans le **chapitre II**, nous présentons les différentes méthodes appliquées dans le domaine de détection de visages

La technique choisie est étudiée dans **le chapitre III** il s'agit des Réseaux de Neurones. nous dressons d'abord un bref historique concernant les travaux sur les réseaux de neurone artificiels. Nous présentons ensuite la structure des réseaux de neurones et leur fonctionnement. Ensuite les topologies des réseaux de neurones sont présentées. Enfin le mode de paramétrage de ces réseaux ainsi que leurs (ou quelques) domaines d'application et leur limitations concluent notre chapitre.

Dans le **chapitre IV**, nous présentons la détection de visages par réseaux de neurones. Il s'agit de détecter le centre des yeux, le bout du nez et les deux coins de la bouche. Pour cette étape l'architecture proposée est un réseau de neurones RBF entièrement connecté.

Enfin dans le **chapitre V**, nous présentons l'implémentation des réseaux de neurones sur la détection de visage et les résultats obtenus.
Finalement nous terminons notre mémoire par une conclusion générale ou l'essentiel des résultats et déductions sont présentés. Nous terminons par la proposition de certaines perspectives que nous jugeons efficaces et utiles.

## I .1 Introduction

Les systèmes biométriques sont généralement divisés en deux catégories :

* La ***biométrie morphologique*** ou ***physiologique***

* La ***biométrie comportementale***.

La *biométrie morphologique* est basée sur l'identification de traits physiques qui sont uniques pour chaque personne et permanents. Dans cette catégorie, on trouve la reconnaissance des empreintes digitales, de la forme de la main, du visage, ainsi que de la rétine ou l'iris de l'oeil.

La *biométrie comportementale*, quant à elle, se base sur le comportement des individus. On retrouve alors les caractéristiques de la signature, l'empreinte de la voix, la démarche ou encore la façon de taper sur un clavier.

Cela fait longtemps que l'on a reconnu l'utilité des technologies liées à la *biométrie morphologique* comme carte d'identité d'un individu. En effet, depuis le 17$^{ème}$ siècle, des chercheurs s'intéressent aux empreintes digitales. C'est Sir Francis Galton (1822-1911) qui démontra que les empreintes digitales sont uniques, et ne changent pas de façon notoire avec le vieillissement des personnes.

Le développement des systèmes biométriques liés à la reconnaissance des formes du visage est des plus récents. En 1982, Hay et Young ont écrit que l'humain utilise, pour reconnaître un visage, les différentes caractéristiques qui le définissent [1]. Grâce à cette remarque des démarches furent entreprises afin de savoir s'il était possible de modéliser de manière informatique ce comportement. Ces recherches donnèrent naissance à plusieurs systèmes de reconnaissance du visage. C'est à partir des travaux de Teuvo Kohonen (1989), chercheur en réseaux de neurones de l'Université d'Helsinki, et des travaux de Kirby et Sirovich (1989) de l'Université Brown du Rhode Island, que fut mis au point par le MIT le système de reconnaissance du visage nommé ***eigenface***.

## I.2 Détection de visage

La détection du visage est la première phase du 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]**.

### I.2.1 Intérêt

On remarque que dans un système biométrique basé sur le visage, la détection est la première phase d'où son plus grand intérêt. Quand on veut réaliser un système biométrique de

reconnaissance de visage, d'identification ou de vérification ; les performances de ce système dépendent fondamentalement de la détection du visage sur une image voir sur une vidéo. Même si les étapes ultérieures du traitement sont aussi importantes et efficaces. En effet la détection du visage doit être d'une grande précision par l'utilisation d'algorithmes et de méthodes ayant fait leurs preuves; de manière à éviter toutes répercutions négatives sur le système biométrique.

### I.2.2 Les difficultés

La détection du visage est une tâche très facile pour l'être humain: l'image est capturée par l'oeil 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].**

- **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.
- **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.
- **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 sur 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.
- **Présence ou pas de quelques composants du visage:** Il s'agit des moustaches, la barbe, des lunettes…qui doivent être prises en considération lors de l'implémentation de l'algorithme.
- **Occultation:** Un visage qui peut apparaitre 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.
- **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.

- **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.
- **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.

## I.3 Les méthodes de détection de visage

Le fonctionnement de la reconnaissance de visage, quelle que soit la technique utilisée, se base toujours sur une capture d'image (vidéo par exemple) ou sur une photo. Les caractéristiques du visage sont ensuite extraites selon la méthode choisie, puis enregistrées dans une base de données.
Il est possible de distinguer quatre approches théoriques[1] :

- **Knowledge-based méthode.** Représentation intuitive du visage : présence de la bouche, des yeux, etc.
- **Feature invariance.** Les traits du visage qui ne changent pas quelque soit les conditions de luminosité, etc.
- **Template matching.** Représentation paramétrique d'un visage qui constitue un modèle, l'étude tourne autour d'une corrélation avec le visage à traiter.
- **Appearance-based method.** Décomposition de l'image en éléments simples (présentation du système Eigenface).

### I.3.1 Knowledge-based Methods ou Méthodes basées sur la connaissance.

Cette méthodologie s'intéresse aux parties caractéristiques du visage comme le nez, la bouche et les yeux. Les positions relatives des différentes composantes du visage sont étudiées après avoir été détectées. La difficulté dans cette approche est de traduire par des règles strictes, à définir, la manière dont le chercheur se représente le visage. Si ces règles sont trop précises, elles

n'identifient pas certains visages (false negative). Dans le cas contraire, elles provoquent de fausses alertes (false positive). Il est évident qu'il est impossible de définir des règles qui tiennent compte de toutes les variabilités comme la position du sujet, par exemple.

Afin d'éliminer de manière progressive les parties de l'image qui ne correspondent pas à un visage, une hiérarchie de méthodes est appliquée. L'objectif est d'appliquer les méthodes précises (et donc coûteuses en temps) qu'aux parties candidates potentielles de l'image.

Selon cette approche, une hiérarchie d'images à résolutions multiples (même image avec des résolutions différentes) est utilisée. En effet, avec un très petit nombre de pixels, il est possible de sélectionner quelques zones images candidates à la représentation d'un visage. Chacune de ces zones sera ensuite étudiée plus précisément (en augmentant la résolution) à l'aide d'histogrammes et de détections de contours. Enfin, la méthode cherche à détecter les composants caractéristiques du visage (yeux). Le taux de réussite de cette méthode reste faible avec surtout un grand nombre de fausses alertes et ne peut donc se suffire à elle-même. Un problème se pose avec cette approche en l'occurrence la difficulté de translater les connaissances du visage d'humain en règles bien définies, ce qui peut provoquer des erreurs de détection et rendre le système pas fiable.

En revanche, elle est une bonne approche pour rapidement se focaliser sur les parties intéressantes de l'image[1].

### I.3.2 Feature-Based Methods, ou recherche d'invariants

L'objectif de cette méthodologie est de trouver les caractéristiques invariantes du visage. Le problème avec cette méthode est que la qualité des images peut être sévèrement diminuée à cause de l'illumination, le Cet algorithme a pour objectif de trouver les caractéristiques structurelles même si le visage est dans différentes position, conditions lumineuses ou changement d'angle de vue. Le problème avec cette méthode est que la qualité des images peut être sévèrement diminuée à cause de l'illumination, le bruit ou l'occlusion.

Cependant, Il existe plusieurs propriétés ou caractéristiques invariables du visage dont les principales sont les suivantes [1].

#### I.3.2.1 Les traits du visage

- Contours

*Algorithme de Sirohey* : Il s'agit d'une méthode de segmentation d'images utilisant une carte de contours et des considérations heuristiques. Il regroupe les contours qui correspondent à ceux du visage et rejette les autres. Son algorithme obtient 80 % de réussite.

- Tâches et rayures

Chetverikov utilise non pas des bordures mais des "tâches" et des séquences linéaires de traits de bordure d'orientation similaire. L'étude se base sur un modèle de visage composé de deux tâches sombres pour les yeux, trois tâches claires pour les pommettes et le nez; les lignes du visage étant caractérisées par les traits d'orientation similaires. La configuration triangulaire des tâches est étudiée ultérieurement.

- Images en nuance de gris

Un filtrage passe-bande est appliqué à l'image, permettant la mise en valeur des composants comme le nez ou la bouche (de gradient caractéristique). A partir de l'histogramme, on peut alors former des seuillages successifs pour délimiter l'image en zones.

- Rapports de brillance

Si les conditions de brillance changent de manière sensible entre deux visages, il n'en est pas de même des rapports de brillance entre deux parties du même visage. En ne gardant que les "directions" (à partir de ces rapports) d'un nombre assez restreint de parties du visage, on obtient un invariant relativement robuste.

- Distances mutuelles

Il s'agit d'une approche probabiliste. Cette méthodologie cherche certaines composantes du visage caractéristiques telles que le nez ou la bouche. Dans un second temps, elle estime les positions relatives entre les éléments grâce à une méthode statistique. Ainsi, il est possible d'éliminer certaines zones de l'image ne correspondant pas à un visage.

- Morphologie

L'objectif est d'extraire des segments analogues à des yeux car il s'agit des composants les plus stables et saillants du visage humain. Ces "eye-analog segments" sont définis comme les contours des yeux :

* Extraction des pixels dont l'intensité change de manière significative,

* Détection des segments qui sont susceptibles d'appartenir à des yeux,

* Considération de combinaisons géométriques pour détecter les autres éléments du visage candidat.

* Vérification par réseau de neurones.

**I.3. 2.2 caractéristique du visage ( Facial facture )**

Cette méthode utilise les plans d'arêtes (Candy detector) et des heuristiques pour supprimer tous les groupes d'arêtes sauf celles qui représentent les contours du visage. Une ellipse est déduite comme frontière entre l'arrière-plan et le visage.

Celle-ci est décrite comme étant formée des « point de discontinuité dans la fonction de luminance (intensité) de l'image ». Le principe de base consiste à reconnaitre des objets dans une image à partir de modèles de contours connus aux préalables. Pour réaliser cette tâche, deux méthodes seront présentées : la transformée de Haugh et la distance de Hausdorff.

- **Transformée de Haugh :** La transformée de Haugh est une méthode permettant d'extraire et de localiser des groupes de points respectant certaines caractéristiques.

Par exemple, les particularités recherchées peuvent être des droites, des arcs de cercle, des formes quelconques, etc. Dans un contexte de détection de visage, ce dernier est représenté par une ellipse dans la carte d'arêtes. L'application de la transformée de Haugh circulaire produirait donc une liste de tous les candidats étant des cercles ou des dérivées.

L'algorithme de base a également été modifié pour voir ainsi apparaitre plusieurs variantes, dont la Randomisé Haugh Transforme, qui peut être appliquée à la recherche de formes quelconques tout comme des cercles ou des ellipses (épx: visages).

Finalement, la transformée de Haugh peut être utilisée pour détecter les yeux et les iris.

Par contre, cette méthode échouera lorsque l'image est trop petite ou lorsque les yeux ne sont pas clairement visibles.

- **Distance de Hausdorff :** cette méthode utilise quant à elle les arêtes comme données de base ainsi qu'un algorithme spécial de Template Machin. En effet, la distance de Hausdorff vise à mesurer la distance entre deux ensembles de points, qui sont la plupart du temps carte d'arêtes (image de recherche) et un modèle.

L'algorithme de base effectue la recherche des meilleurs endroits de correspondance partout dans l'image (translation) et ce pour différentes rotations. Cette recherche peut également inclure un facteur d'échelle afin de détecter des variations du modèle. Cette méthode a été utilisée avec succès pour la détection de visages avec vues frontales. L'adaptation de celle-ci amène cependant certains problèmes, car différents modèles devraient une étape complexe de décision aurait pour mission de départager les fausses détections ainsi que les détections multiples.

### I.3.2.3 Texture

La texture de l'être humain est distincte et peut être utilisée pour séparer les visages par rapport à d'autres objets. Augustin et Soufraont développé une méthode de détection de visages sur une image en se basant sur la texture.

La texture est calculée en utilisant les caractéristiques de second ordre sur des sous images de 16× 16 pixels. Il est possible d'identifier un visage à partir de sa texture. En effet, trois types de textures sont utilisées : la peau, les cheveux, et "les autres". Les textures sont calculées sur des masques de 16*16 pixels. Puis, un réseau de neurones réalise une corrélation en cascade dans le but de classifier de manière supervisée ces textures. Aucun test n'a encore été réalisé pour la localisation de visages.

### I.3.2.4 Couleur de la peau

La couleur de la peau de l'être humain a été utilisée pour la détection des visages et sa pertinence a été prouvée comme caractéristique spécifique au visage. Cependant la couleur de la personne est différente selon les personnes et leur origine. Plusieurs études ont démontré que la plus grande différence s'étend largement entre intensité plutôt que la chrominance.

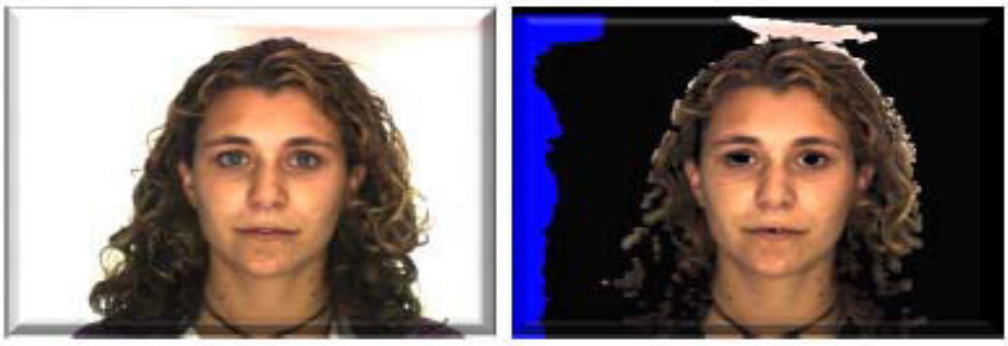


**Figure I.1 :** La détection de la couleur de peau.

### I.3.2.5 Caractéristiques multiples ( Multiple filature )

Au terme de recherches récentes plusieurs méthodes combinent les différentes caractéristiques faciales pour la détection des visages. La plupart utilisent des propriétés globales comme la couleur de la peau, la taille et la formes du visage pour trouver les candidats, puis les vérifier localement en se basant sur les caractéristiques détaillées comme les yeux, les sourcilles, le nez et les cheveux.

L'approche type commence par détecter les régions qui représentent la couleur de peau.

Puis regrouper les pixels définis comme peau en utilisant « connected component analysis » ou « clustring algorithm ». Si la région regroupée représente une ellipse ou une forme ovale, elle est alors considérée comme candidat, auquel on appliquera les caractéristiques locales pour la vérification.bruit ou l'occlusion.

### I.3.3 Template-matching ou appariement de gabarit.

Le *template matching* [1], ou l'appariement de gabarit, est une des techniques de détection de visage la plus simple de mise en oeuvre. Elle consiste en la comparaison d'intensité de pixels entre un gabarit prédéfini et plusieurs sous-régions de l'image que l'on désire analyser. Cette méthode s'effectue par plusieurs balayages sur l'ensemble de l'image. Les endroits les plus propices à la présence de visages sont identifiés par une différence minimale entre le gabarit et l'image.

Deux désavantages majeurs restent à déplorer. En effet, les performances du *template matching* diminuent avec les différentes expressions du visage. L'échelle de la phot est aussi un facteur limitant. Pour pallier à ce problème, divers gabarits peuvent être définis, mais la gestion des résultats obtenus peut s'avérer complexe. Ceci implique donc un balayage de l'image à diverses échelles, ainsi que le filtrage des multiples détections appartenant aux mêmes visages. De ce fait l'utilisation d'un gabarit plus ou moins adapté au type d'objet recherché peut nuire à une détection efficace et diminuer la précision des résultats.

La localisation des différentes caractéristiques du visage peut être déduite à partir des positions correspond La détection des visages entiers ou des parties de visage se fait à travers un apprentissage d'exemples standards de visages. La corrélation entre les images d'entrées et les exemples enregistrés est calculée pour la décision.

#### I.3.3.1 Faces prédéfinies ( Prédéfinie Face Template )

La détection frontale des visages la plus récente a été rapportée par Sakai et al. Les sous-gabarits souvent utilisés sont les yeux, le nez, la bouche et les contours du visage. Chaque sous-gabarit est défini par une segmentation des lignes. Tout d'abord une comparaison est effectuée entre les sous-images et les contours de gabarit pour détecter les visages candidats. Puis on translate les sous-gabarits comme les yeux, le nez … sur les positions candidats. En résumant la détection s'effectue en deux étapes : la première est la détection des régions candidates, la deuxième est l'examen des détails pour déterminer les caractéristiques du visage.

### I.3.3.2 Les Templates déformables ( Formable Template )

Cette méthode a été utilisée par Yuille et al. Pour modéliser les caractéristiques faciales qui s'adaptent élastiquement par rapport au modèle du visage. Dans cette approche, les caractéristiques faciales sont décrites par des gabarits paramétrés. Une fonction est définie pour relier les contours, les sommets et les angles dans l'image d'entrée, pour faire correspondre les paramètres sur les gabarits. La meilleure adaptation du modèle élastique est de trouver la fonction énergétique en minimisant les paramètres.. Celles-ci sont déterminées au préalable manuellement en positions relatives par rapport aux dimensions du gabarit. Seulement, il convient de noter que le gabarit peut ne pas être parfaitement positionné en translation, en échelle et en rotation sur le visage à détecter. Dans ce cas, les coordonnées déduites sont légèrement erronées.

### I .3.4 Appearance-Based Methods ou methods basées sur l'apparence : Eigenface

La différence entre cette méthode et Template matching est que les modèles (Template) sont lus à partir des images d'apprentissage qui doivent être représentatives et à différentes positions du visage.

En général, appearence-based method s'appuie sur des techniques d'analyse statistique et d'apprentissage automatique pour trouver des caractéristiques significatives des visages et des non visages. L'apprentissage est organisé par le biais de modèles de distributions ou par une fonction discriminante.

La réduction de dimension est habituellement accomplie par égard à la capacité d'estimation et l'efficacité de détection.

Plusieurs méthodes basées sur l'apparence peuvent être interprétés dans un cadre probabiliste. Une image ou un vecteur caractéristique est dérivé d'une image considérée comme une variable aléatoire X, cela représente la caractéristique du visage ou du non visage[2].

### I.3.4.1 La méthode Eigenface

Un L'analyse en composantes principales(PCA) a été présenté par MP .Pentland et MA. La PCA n'a besoin d'aucune connaissance de l'image, son principe de fonctionnement est la construction d'un sous espace vectoriel retenant que les meilleurs vecteur propres, tout en gardant beaucoup d'information utile non redondante (très efficace pour réduire la dimension des données).

Une des technologies de reconnaissance de visages les plus connues est développée et supportée par le MIT et se nomme "*Eigenface*". Ce système décompose l'image en une série d'images teintées de nuances de gris, chacune mettant en évidence une caractéristique particulière

du visage. Les zones claires et foncées ainsi créées forment les caractéristiques uniques du visage et ce sont elles que l'on nomme *eigenface*. On en extrait ainsi de 100 à 125 par visage.

L'élément essentiel de cette méthode repose sur la réalisation d'une analyse en composantes principales (ACP) sur une série d'image représentant les visages à détecter. Cette réduction de dimension fournie les premiers vecteurs propres (*eigen*) représentant les plus fortes différences entre les points que nous analysons : nez, yeux, etc. L'ACP permet donc de distinguer une image contenant un visage d'une image n'en contenant pas. Seulement, elle n'est pas capable de reconnaître deux visages identiques mais d'expressions différentes.

Tout comme le *template matching*, l'étape de détection de l'*Eigenface* consiste à effectuer un balayage de la zone à traiter. Ensuite, une reconstruction avec les premiers vecteurs propres est réalisée pour chacune des imagettes à analyser. Si l'image reconstruite possède suffisamment de points communs avec l'imagette d'origine, alors la sous-image possède les caractéristiques discriminantes de la classe générée et peut être examinée correctement avec la base générée par les vecteurs propres. La distance entre ces deux images est calculée à l'aide d'une métrique particulière telle que celle utilisée pour l'appariement de gabarit[2].

**Figure I.2 :** Images utilisées pour l'apprentissage et visages propres correspondants.

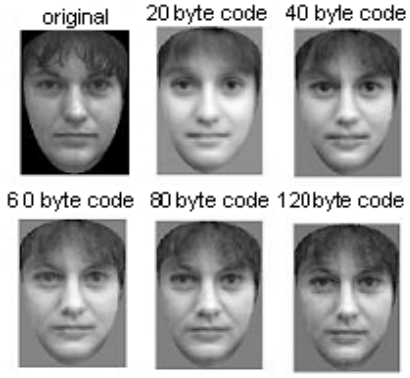



**Figure I.3 :** Visage reconstruit à partir de visages propres.

Le visage original ne faisait pas partie de la base de données utilisée pour la construction de ces visages propres.

Cette méthode est robuste aux variations d'éclairage. En revanche, elle ne l'est pas du tout aux variations d'échelle (mais un changement d'échelle préalable est toujours possible). Plus embêtant, cette méthode n'est pas robuste aux variations de l'expression du visage. Ainsi, Eigenface souffre relativement des même inconvénients que ceux inhérents aux *template matching*, ce qui ne facilite pas son utilisation.L'exemple Le plus récent de l'utilisation d'eigenvectors est dans la reconnaissance des visages de Kohonen dans laquelle un simple réseau de neurone a démontré sa performance dans la détection des visages passant par la normalisation des images de visage.

Le réseau de neurones estime la description des visages par l'approximation de vecteurs propres et l'auto corrélation matriciel de l'image.

Kirby and sirovich ont démontrés que les images de visage peuvent être codé linéairement en nombre modeste d'image d'apprentissage. La démonstration est basée sur la transformée de Karhumen-loeve, appelée aussi « analyse des composants principales » ou « la transformée de Hottlling ».

La première idée proposée par Pearson en 1901 et par Hotteling en 1933 est d'avoir une collection d'image d'apprentissage de n×m pixels représentés comme des vecteurs de taille m×n, le principe des vecteurs est de trouver le sous-espace optimal pour ajustement de l'erreur entre la projection des images d'apprentissage dans ces sous-espaces et les images originales miniaturisées.

**I.3.4.2 Distribution based ou méthode basée sur la distribution.**

Sung et Poggio ont développé la méthode basée sur la distribution pour la détection du visage, elle démontre comment la distribution de l'image appartenant à une seule classe d'objet peut être classifiée comme exemple de classe positive ou négative. Ce système est constitué de deux composants, le premier est constitué de modèle d'exemple basé sur la distribution de visage et de non visage, le deuxième est un classifieur à perceptron multicouche. Chaque exemple de visage et de non visage est tout d'abord normalisé, puis redimensionné l'image (19×19 pixels) pour la traiter comme 361 vecteurs dimensionnels, ces structures sont regroupés en six bouquets de visages et de non visages en utilisant l'algorithme modifié *K-means[2].*

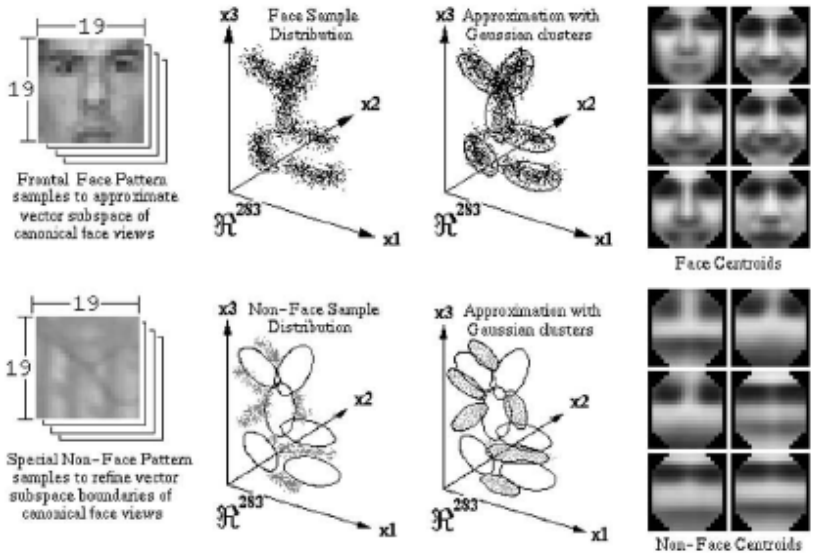



**Figure I.4 :** Les groupes de visage et non visage utilisés par Sung Poggio .

Chaque classe est représentée comme une fonction gaussienne multidimensionnelle avec la moyenne de l'image et la covariance matricielle.

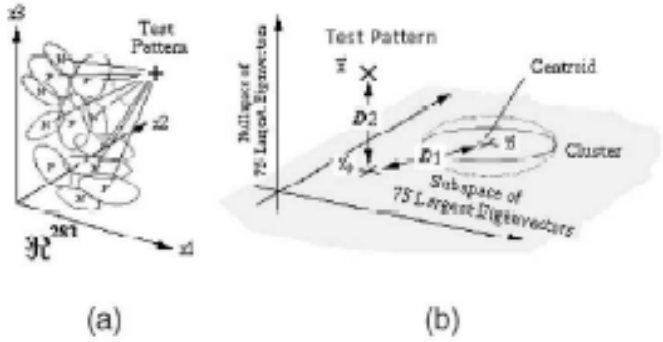



**Figure I.5 :** Les mesures de distance utilisées par Sung Poggio .

Deux distances métriques sont calculées entre le modèle de l'image d'entrée et les prototypes des classes, la première distance est la normalisation de Mahalanobis entre les modèles de tests et les bouquets, .la deuxième est la distance Euclidienne entre les modèles de tests et leurs projections dans un sous-espace à 75 dimensions. La dernière étape consiste à utiliser un Réseau Perceptron Multicouche (MLP) pour classifier les fenêtres des visages et des non visages.

### I.3.4.3 Réseaux de neurones

Nous utilisons le réseau de neurones pour classifier les pixels de l'image, en tant que visage ou non-visage. Dans toute utilisation de réseaux de neurones, il faut définir une topologie du réseau. Il n'y a aucune règle pour définir cette topologie et c'est souvent par tests successifs qu'une bonne topologie est définie.

Un visage se distingue en effet surtout par des yeux, un nez et une bouche. La topologie de base sera donc d'une unité finale fournissant une réponse binaire ou probabiliste. On mettra derrière cette unité les couches cachées du réseau, on appelle notamment cela une topologie de base car le nombre d'unités, leur taille et leur position restent non empiriques et ne peuvent en conséquence pas être fermement fixés. Le nombre de couches de chaque unité peut être augmenté utilisant des doubles et triples couches.

Les réseaux de neurones les plus répandus et les plus simples à la fois restent les perceptrons multicouches (PMC) qui consistent en une succession de 9 couches, interconnectées totalement ou partiellement.

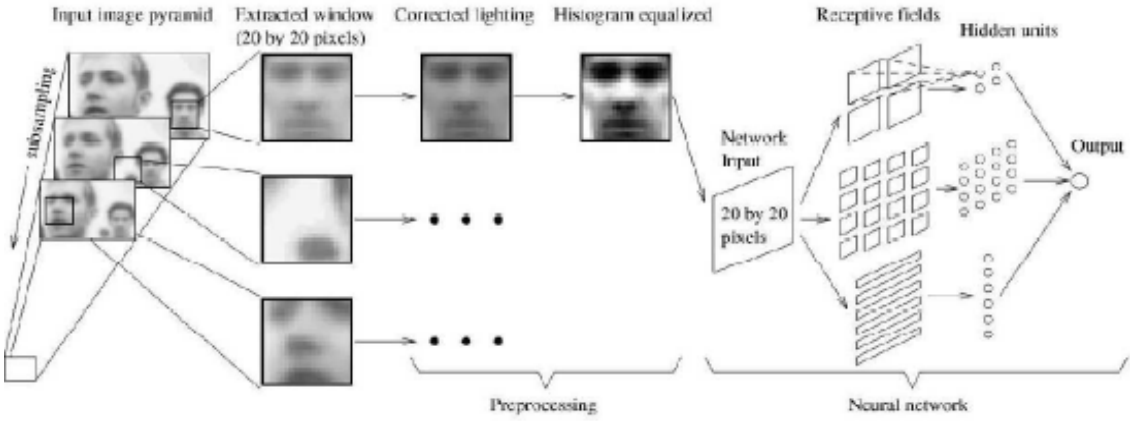



**Figure I.6 :** Diagramme de la méthode de Rowley .

L'algorithme d'apprentissage total reviendra à transmettre à tous les PMC, traitant chacun une unité, le résultat attendu. Si l'exemple à apprendre est un visage, la première étape est de transmettre aux PMC les yeux, la bouche et le nez. De là, chaque PMC applique son algorithme d'apprentissage.

L'inconvénient de cette approche réside dans le temps de calcul qui ne permet pas souvent de faire des traitements en temps réel.

**I.3.4.4 Naïve Bayes classifier ou classifieur naïf de Bayes**

A la différence des méthodes basées sur l'apparence globale des visages, Shneiderman et Kanade ont décrit à Naïve Bayes Classifier pour estimer la probabilité de l'apparence locale et la position des composants du visage en multiple résolution. Il y a deux raisons pour l'utilisation de la méthode Naïve Bayes Classifier :

- Premièrement une meilleure estimation de la fonction conditionnelle de densité des sous-images,
- la deuxième raison est que Naïve Bayes Classifier fournie une fonctionnalité supplémentaire après la probabilité, ça consiste à capturer la jointure statistique de l'apparence locale et la position des objets.

A chaque échelle, l'image est décomposée en quatre sous-régions rectangulaires, chaque sous-région est projetés dans un espace dimensionnel inférieur utilisant la PCA en l quantifiant dans une infinité d'arrêtes, puis on aura l'estimation statistique des sous-régions à partir des projections simples jusqu'à encoder l'apparence locale. Sous cette formulation, la méthode décide de la présence d'un visage quand le coefficient de ressemblance est supérieur au coefficient de probabilités antérieures.

**I.3 .4.5 Support vector machine (SVM)**

La méthode SVM fait partie de la gamme des méthodes à apprentissage statistique, proposé par V .Vapnik en 1995. Le système de détection est conçu avec des images de formation et aucune connaissance préalable n'est requise sur l'image.

Le principe est de définir le vecteur caractéristique d'un visage comme un ensemble des valeurs de luminance choisies sur une fenêtre de taille prédéfinie, puis apprendre les classes de visage et de non visage. Le choix de la séparation se caractérise par une hyper-surface qui classifie correctement les données et qui se trouve le plus éloigné possible de tous les exemples.

Support vector machine est une des premières méthodes appliquée a la détection de visage par Osuna et al. SVM peut être considéré comme un nouveau modèle de classifieur d'apprentissage de fonction polynomial, réseau de neurone ou radial basis function(RBF) .la plupart des classifieur d'apprentissage (Bayesian, Réseau de Neurone, RBF) sont basés sur la minimisation de l'erreur d'apprentissage « l'erreur empirique », SVM opère avec un autre principe appelé « structural risk minimisation » qui a pour but de minimiser les saut supérieurs sur les erreurs généralisées probables.

**I.3.4.6 Hidden Markov Model (HMM) ou modèle de Markov caché**

L'hypothèse sous jacentes de Hidden Markov Model (HMM) est que les structures peuvent être caractérisées par des processus à paramètres aléatoires qui peuvent être estimés avec précision,

d'une manière bien définie. Lors du développement de la méthode HMM pour la reconnaissance des formes, de nombreuses conditions cachées doivent être clarifiées pour former le modèle. Puis on pourra instruire HMM à apprendre la transitionnelle probabilité entre les états à partir des exemples qui seront représentés comme une séquence d'observation.

Le but derrière l'apprentissage du HMM est de maximiser la probabilité de l'observation en ajustant les paramètres HMM avec les méthodes de segmentation standard Viterbi et l'algorithme de Baum-welch. Après l'apprentissage du HMM, la probabilité de sortie détermine à quelle classe l'entrée observée appartient.

Intuitivement, les parties du visage sont divisées en plusieurs régions comme le front les yeux, le nez, la bouche et le menton .alors un processus peut reconnaitre chaque partie du visage qui sont observe dans un ordre approprié (du haut vers le bat et de la droite vers la gauche).

Au lieu de s'appuyer sur l'alignement exact comme dans le Template matching ou les méthodes basées sur l'apparence (les caractéristiques du visage comme les yeux le nez ont besoin d'être bien aligné avec les points de référence), cette approche a pour objectif d'associer les régions du visage avec les états de la densité continue Hidden Markov Model.

La méthode HMM est généralement utilisée pour traiter les parties du visage comme une séquence d'observation de vecteur et chaque vecteur est un groupe de pixels.

Durant l'apprentissage et les tests, l'image est scannée dans un certain ordre (généralement du haut vers le bat) et l'observation se fait sur des blocs de pixels. La frontière entre les bocs de pixels est représentée par les transitions probabilistes entre les états. Les données de la région de l'image sont modelées selon la distribution Gaussien multi variable.


**Figure I.7 :** La chaine de Markov pour la localisation du visage .

#### I.3.4.7 Information-Technical Approach

Les propriétés spatiales des caractéristiques du visage peuvent être modélisées à travers plusieurs aspects. Un des moyens puissants est les contraintes contextuelles qui sont souvent appliquées à la segmentation de texture.

Les contraintes contextuelles pour les caractéristiques du visage sont habituellement spécifieés par un petit voisinage de pixels. La théorie Markov Random Field(MRF) fournie une voie commode et consistante pour le modèle contexte-dépendance des entêtes, par exemple les pixels d'une image et les caractéristiques corrélées. Cette dernière s'obtient par la caractéristique de l'influence mutuelle entre les différentes entités utilisant la distribution conditionnelle MRF.

## I.5 Approche retenue

Parmi toutes les techniques présentées lors de la section précédente, certaines s'avèrent plus efficaces ou rapides que d'autres.

En comparant ces différentes méthodologies, nous nous sommes aperçu qu'une majorité d'entre elles ne peuvent pas détecter des points précis du visage tels que les yeux.

Par contre, certaines fonctionnent très bien pour la détection du visage entier. Il serait donc intéressant de pouvoir améliorer la détection des zones critiques du visage.

Après avoir comparé et analysé ces différentes méthodes, l'utilisation d'une méthode hybride alliant les avantages de certaines techniques présentées précédemment à de nouvelles approches a été retenue. Cette solution est composée des étapes suivantes[2] :

(1) Passage de l'image en niveau de gris.

(2) Traitement du contraste de l'image.

(3) Conception des cartes repérant les régions d'intérêts.

(4) Détection des visages par réseaux de neurones.

(5) Validation du visage trouvé.

### I.5.1 Type de photos utilisées

Un visage consiste en la surface d'un solide partiellement déformable. Pour une même personne l'image projetée dépend principalement des paramètres suivants[2] :

- Paramètres liés à la personne

  Instantanés (peuvent être modifiés pour la prise de vue) : expression, accessoires (lunettes, chapeau).

Temporaires (ne peuvent être modifiés pour la prise de vue) : âge, capillarité.

- Paramètres liés à l'acquisition

Instantanés : éclairage, pose.

Temporaires : paramètres de la caméra.

Idéalement, il serait intéressant que notre système d'identification puisse reconnaître un individu quelques soient les paramètres. Seulement, dans un premier temps, afin de simplifier l'image sur laquelle on a tendance à travailler, pour cela on fixe les paramètres instantanés.

De ce fait, toutes les photos sont prises sous un même éclairage de face, dans des conditions simples : pas de lunettes, ni d'accessoires particuliers. On se base donc sur des images "simples" comme des photos d'identité.

## I.6 Conclusion

La détection et la reconnaissance d'individus demeurent des problèmes complexes,malgré les recherches actives actuelles. Il y a de nombreuses conditions réelles, difficiles à modéliser et prévoir, qui limitent les meilleurs systèmes actuels.

Le système de détection proposé présente une alternative simple mais robuste. La détection est réalisée par une méthode hybride alliant traitement de l'image et réseau de neurone.

Les résultats de la première phase de détection sont fiables sur les images simples que nous considérons. Seulement, dès que l'image d'entrée devient plus complexe, le taux de mauvaise localisation augmente. Il serait donc intéressant de pallier à cette limite en affinant les traitements préalables afin d'augmenter leurs précisions (amélioration de la détection de contours, etc…)

## II.1 Introduction

Dans la littérature, plusieurs algorithmes de reconnaissance du visage ont été crées pendant ces dernières années. L'objectif est de montrer la complexité des méthodes pouvant être utilisées dans de telles applications. La performance de ces algorithmes dépend fortement de la qualité des résultats de détection et de normalisation des visages. Cela veut dire que: plus la précision obtenue est élevée, plus les conditions ( d'acquisition ,d'éclairage, pose etc.) se rapprocheront de celles de la phase d'apprentissage, ce qui donne une authentification efficace.

En effet, une mauvaise localisation et/ou normalisation du visage cause une chute drastique du taux de reconnaissance et rend le système d'authentification moins performant.

Pour avoir un système d'authentification plus performant, il faut que le module de prétraitement soit le plus précis possible afin d'obtenir les meilleures conditions initiales.

L'étape de prétraitement joue le rôle d'une réduction des données ainsi d'une atténuation des effets d'une différence de conditions lors des prises de vues. Dans notre travail nous supposons que les images sont prises dans les conditions favorables suivantes:

1- Une vue frontale de toutes les images.

2- L'éclairage des visages ne change pas.

3- Une distance fixe entre le visage et la caméra (plus/moins quelque cm).

## II.2 La détection des visages

### II.2.1 détection de visages dans une séquence d'images fixes

Dans la littérature, plusieurs travaux portant sur la détection de visage et de mouvement sont souvent utilisés dans des applications de codage, de poursuite, d'indexation, de surveillance, etc.

Parmi les méthodes existantes, on cite :

- L'approche la plus répandue est sans doute, celle utilisant **un réseau de neurones** pour classifier les pixels de l'image, en tant que visage ou non-visage.

L'inconvénient de cette approche réside dans le temps de calcul qui ne permet pas souvent de faire des traitements en temps réel.

-L'approche qui consiste à déterminer des **points d'intérêts** ( **maxima locaux** d'un filtre gaussien aux dérivées secondes), en partant de ces derniers, on réalise une détection de contours qui seront examinés pour être groupés en régions. Le groupement est basé sur leur proximité et leur similarité en orientation et épaisseur. A partir de chaque région, on définit alors un vecteur dont on calculera la moyenne et la matrice de covariance par rapport aux différents vecteurs modèles. Le

critère d'appartenance à un élément du visage s'appuie sur la distance de Mahalanobis, les différents candidats sont alors groupés en se basant sur un modèle de connaissance indiquant leur position relative. Chaque composant du visage est enfin analysé avec un réseau Bayes en. L'intérêt de cette méthode est qu'elle peut détecter des visages dans diverses poses.

-Des études ont montré que ***la variabilité de la couleur de la peau*** tenait plus de la différence d'intensité plutôt que de la chromaticité. On peut utiliser plusieurs types d'espaces de couleurs (RGB, RGB normalisé, HSV, YcrCb, YIQ, YES, CIE XYZ, CIE LUV). L'information de couleur est un outil efficace pour identifier des zones du visage. Cependant, cela n'est pas utilisable lorsque le spectre de couleurs varie de manière significative entre l'arrière plan et le visage.

-D'autres études ont toutefois montré que la combinaison **d'analyse de formes**, de **segmentation de couleurs** et d'**informations de mouvement** pour la localisation et le suivi de visages dans des séquences d'images conduisant à des résultats intéressants.

On décrit un algorithme de segmentation (à travers une suite d'images) pour l'extraction des visages dans un environnement inconnu, tout en respectant les contraintes d'environnement de l'application.

L'importance de l'approche proposée réside dans sa simplicité et son efficacité, elle est basée sur l'exploitation d'un algorithme de détection de visage associé à la segmentation par le mouvement et à des techniques de traitement (ACP, interpolation, modèle 2D) pour aider la phase décisionnelle (algorithme B-snake) à converger d'une manière plus efficace, stable et rapide.

La démarche adoptée consiste à détecter, tout d'abord, une zone d'intérêt par la technique de teinte chair ou la segmentation de mouvement-couleur suivant un critère de pertinence bien défini. Ensuite, les paramètres les plus importants sont évalués en utilisant une méthode itérative basée sur une analyse par composantes principales.

Plusieurs vues sont généralement utilisées pour la mise au point. Les autres paramètres sont ensuite déterminés grâce à l'algorithme de suivi temporel.

La seconde étape consiste à comparer la projection du modèle avec les données extraites de l'image 2D. Cette comparaison est faite en se basant sur deux types de primitives : des points d'intérêt (correspondant aux sommets du modèle par exemple) ou des segments de droites (correspondant par exemple aux arêtes). À chaque primitive est ensuite associée une mesure permettant de savoir si la pose est en accord ou non avec l'image 2D.

Enfin, une méthodologie d'interpolation nous permet de décider du contour approximatif de l'objet (visage) à détecter. La méthode de segmentation B-snake prend en compte la topologie de la segmentation dans la phase de la localisation prédéterminée.

## II.3 Détection du visage dans une image 2D

Selon Hjelmas et Low [4], les méthodes de détection de visages peuvent être classées en « approche globale » dans laquelle on analyse le visage dans son entier, ou en « approche locale », dans laquelle on va essayer de détecter, localiser et regrouper les différents éléments constitutifs du visage : nez, yeux, bouche,… Des informations complémentaires peuvent aussi être utilisées pour les différentes détections de visage ce qui rentre dans la catégorie des méthodes globales. Elles diffèrent seulement dans leur première phase de mise en oeuvre, comme l'illustre la figure 1, c'est-à-dire en fait par la technique utilisée pour caractériser l'image à traiter. La première méthode utilise les propriétés géométriques à travers le calcul des moments de Zernike [5]. La seconde est basée sur une projection sur un sous espace nommée « Eigen faces » c'est une technique bien connue en particulier dans le domaine de la reconnaissance de visages [6]. La deuxième étape de ce processus de détection est assurée par un réseau de neurones entraîné par les vecteurs caractéristiques calculés à partir des « Eigen faces » ou des moments de Zernike. Sa sortie délivre des points de contour supposés délimiter un visage dans l'image d'origine.

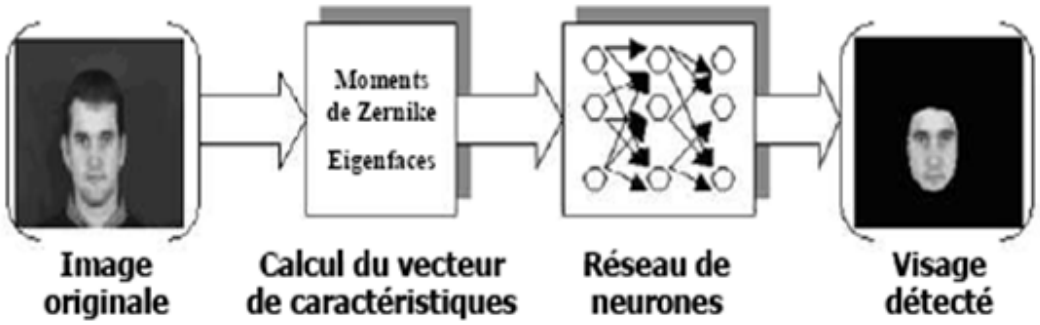



**Figure II. 1 :** Schéma de principe de la méthode de détection du visage

### II.3.1 Présentation des moments de Zernique

Les moments de Zernike ont été introduits en 1934 [5]. Dans le domaine du traitement de l'information, les moments de Zernike ont beaucoup été utilisés pour leur propriété d'orthogonalité qui permet la génération de descripteurs non redondant et leurs propriétés d'invariance en translation, en échelle et en rotation. Ainsi, on retrouve les moments de Zernike dans beaucoup de travaux concernant la reconnaissance d'images de personnes [7], l'indexation d'images dans les bases de données, l'analyse et la description de forme d'objet 2D ou 3D… .

### II.3.2 Introduction aux « Eigen faces »

Les « Eigen faces » furent le premier type de caractérisation utilisé avec succès dans des traitements faciaux tels que la détection et la reconnaissance du visage [6]. Cette méthode est basée sur la décomposition de l'image traitée selon plusieurs directions de variation autour d'une image moyenne.

### II.3.3 Définition d'une mesure de qualité de la détection

Pour mesurer l'efficacité d'un algorithme de détection du visage, nous proposons de mettre en oeuvre une mesure quantitative basée sur le rapport entre la surface de visage détecté et la surface de visage présent dans l'image originale. Pour permettre de réaliser une telle mesure, l'ensemble des images de la base de test ont été divisées en trois régions (cf. figure 2). La région blanche contient les *W* pixels jugés essentiels dans la détection du visage (nez, yeux, bouche,…). La région grise contient des pixels éléments du visage mais jugés non essentiels pour la bonne détection de celui-ci. La région noire contient les *B* pixels non éléments du visage.

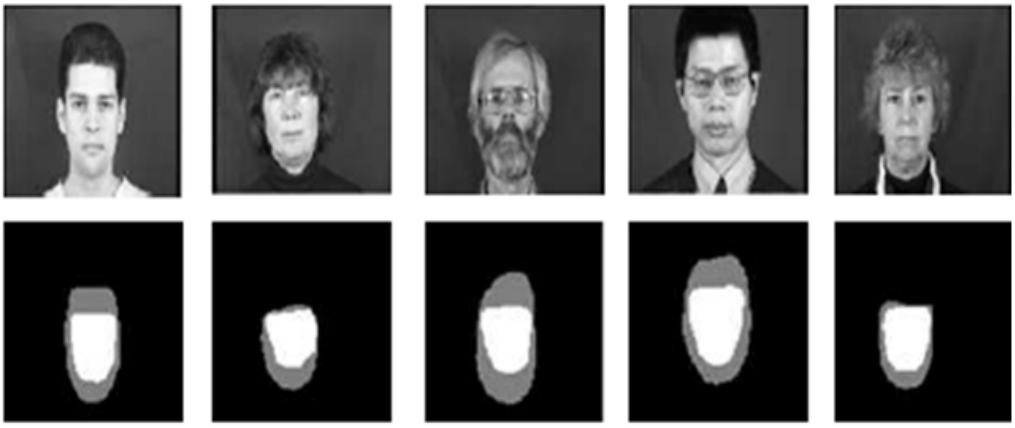


**Figure II.2 :** Images originales et les trois zones caractéristiques associées à chacune d'elles

La région blanche doit être impérativement contenue dans le contour résultant de l'application de l'algorithme de détection. La détection de pixels de la région grise est considérée comme optionnelle et n'influera pas sur le résultat de la mesure. Par contre aucun pixel contenu dans la région noire ne devrait être détecté, par conséquent, si un pixel élément de cette région est détecté, cela pénalisera le résultat de la mesure.

Le critère de mesure proposé est basé sur le calcul des deux quantités nommées Gdr (Good Detection Rate) et Qdr (Quality Detection Rate) définies par les équations (II.1) et(II.2) :

$$G\,dr = \frac{W_1}{W}.100 \qquad \text{(II.1)}$$

$$Q\,dr = \left(\frac{W_1}{W} - \frac{B_1}{A-B}\right)100 \qquad \text{(II.2)}$$

*W1* et *B1* représentent respectivement le nombre de pixels correctement et faussement détectés comme faisant partie du visage et *A* le nombre total des pixels de l'image d'origine. Gdr mesure à quel point les pixels composant l'essentiel du visage ont été correctement détectés. Qdr donne une mesure plus stricte prenant en compte le nombre de pixels faussement détectés comme appartenant au visage dans le calcul de la valeur quantitative de détection. Ces deux mesures sont complémentaires. En effet, si on connaît seulement Gdr on ne possède aucune information sur le taux de pixels attribués à tort au visage. Par ailleurs, la seule connaissance de Qdr ne donne pas d'information sur le nombre de pixels détectés appartenant à l'essentiel du visage et qui ne sont pas inclus dans la région détectée comme visage. En conclusion l'algorithme de détection sera d'autant meilleur que Gdr sera élevé et Qdr proche de Gdr (puisque Qdr est inférieur ou égal à Gdr).

## II.4 Définition d'une stratégie de reconnaissance

Les "Eigen faces" et les "Fisher faces" sont deux méthodes fondamentales dans les approches proposées pour la reconnaissance de visages. Toutes deux sont basées sur la décomposition de l'image sur un sous espace réduit et sur la recherche d'un vecteur optimal de caractéristiques décrivant l'image du visage à reconnaître.

Dans ces méthodes, une image de taille *p* par *q* pixels, est représentée par un vecteur de taille *n* ($n = p \times q$) dans un espace de grande dimension. Les images de visages sont linéarités comme des vecteurs *x* puis sont exprimées en tant que combinaison linéaire de la base orthogonale Φ réduite à une dimension choisie m :

$$\Phi_i : x = \sum_{i=1}^{n} a_i \phi_i \approx \sum_{i=1}^{m} a_i \phi_i \qquad \text{(II.3)}$$

### II.4.1 Analyse en Composantes Principales ou PCA

Le premier système de reconnaissance de visages qui a permis d'obtenir des résultats significatifs a été réalisé par Turks et Portland [6] en utilisant la méthode dite des « Eigen faces ». La base orthogonale Φ définit un espace appelé « fac espace » de dimension *m* inférieure à la taille des vecteurs images ($m << n$) en résolvant l'équation **(II.4)** :

$$C\Phi = \Phi A \qquad \text{(II.4)}$$

$$\Phi = [\phi_1, ....., \phi_m]^T$$

où *C* est la matrice de covariance du vecteur image *x*:

$$C = (x - x).\overline{(x - x)}\,\mathrm{T} \qquad \text{(II.5)}$$

est la matrice des vecteurs propres appelés aussi eigenfaces de la matrice de covariance *C*, et Λ est la matrice des valeurs propres associées.

### II.4.2 Analyse Discriminante Linéaire ou LDA

Belhumeur et col. [8] ont proposé la méthode des "Fisherfaces" pour résoudre le problème de la robustesse face aux variations de pose, d'illumination et d'expressions mettant en difficultés la PCA. Le principe consiste à appliquer l'analyse discriminante linéaire (LDA) pour sélectionner le sous espace linéaire Φ qui maximise le rapport

$$\frac{\left|\Phi^T S_b \Phi\right|}{\left|\Phi^T S_W \Phi\right|} \qquad \text{(II.6)}$$

$$\text{Ou : } S_b = \sum_{i=1}^{M} (\bar{x}_i - \bar{x})(\bar{x}_i - \bar{x})^T \qquad \text{(II.7)}$$

est la matrice de variance inter-classe

$$S_W = \sum_{i=1}^{M} \sum_{xeX_I} (x - \bar{x}_i)(x - \bar{x}_i)^T \qquad \text{(II.8)}$$

est la matrice de variance intra-classes

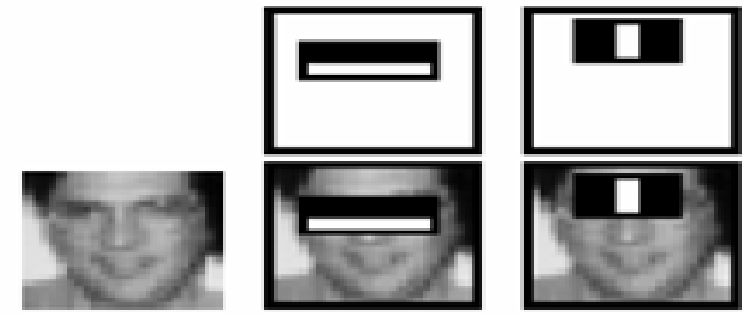
M étant le nombre d'individus de la base de données pour construire l'espace de projection Φ. Les Fisherfaces tels que proposés dans [8] et [9] commencent par appliquer une analyse en composantes principales (PCA) pour réduire la dimension du sous-espace de projection avant de mettre en oeuvre l'analyse discriminante linéaire (LDA). Ceci permet de résoudre le problème de singularité de la matrice de variance intra-classes.

## II.5 Détection sur des images extraites par l'algorithme de Viola Jones

### 1) Principe

Cette méthode utilise des classifiées dits de « Haïr ». Ce sont des masques de la forme de deux ou plusieurs rectangles . Pour calculer la réponse d'une image à ces masques on effectue la somme des pixels de l'image présents sous le rectangle blanc du

classifieur moins la somme des pixels de l'image sous le rectangle noir. Ces classifieurs sont dits faibles dans le sens où ils donnent un résultat de détection que légèrement meilleur que le hasard. Cependant l'association de plusieurs de ces classifieurs, sélectionnés et associés lors d'une phase d'apprentissage permet d'obtenir un classifieur dit fort

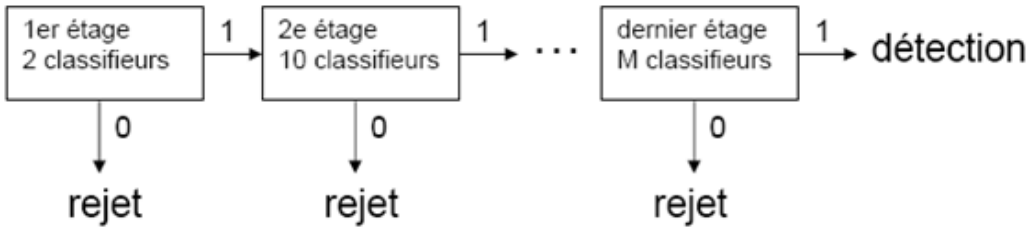


.

**Figure II .3** Deux exemples de classifier ainsi que l'application donnant une réponse maximum

Le premier classifiera utilise la déférence d'intensité entre la zone des yeux et celle des joues alors que le deuxième se sert de la variation d'intensité entre les yeux Pour obtenir un classifier performant il est nécessaire d'associer une centaine de ces classifier de Haïr ce qui oblige à scanner une image avec tout ces classifiées pour différentes échelles (car l'on ne connaît pas à l'avance la taille du visage dans une image) ce qui demande beaucoup trop de calculs. Pour résoudre ce problème, Viola et Jones ont eu l'idée d'utiliser une structure en cascade. Ils cherchent d'abord un visage en utilisant une association de 2 classifiées et pour chaque réponse positive il teste une

association comportant plus de classifiées. Les premiers étages sont paramétrés pour avoir un taux de détection de 100% en contre partie d'un taux de fausse alarme important. Les derniers étages permettant de diminuer ces fausses alarmes




**FigureII.4:** Structure en cascade proposée par Viola Jones

Un visage ne peut être détecté que si la zone traitée est validée par tous les étages de la structure. Lorsqu'il n'y a pas de visage, la détection est rejetée dès les premiers étages n'utilisant ainsi qu'une dizaine de classifier d'où une réduction importante des calculs effectués.

**2) Images extraites par l'algorithme de Viola Jones**

En appliquant cet algorithme sur l'ensemble de notre base de données (images de gauche) on obtient les visages de droite :

On remarque quelques variations dans la position des yeux, ou le facteur d'échelle entre deux visages extraits par cet algorithme

**Figure II .5**: Détection de visage par l'algorithme de Viola Jones

On remarque quelques variations dans la position des yeux, ou le facteur d'échelle entre deux visages extraits par cet algorithme

## II.6 Conclusion

Nous avons présenté dans ce chapitre la localisation de régions d'intérêts de visage et l'authentification d'individus, ainsi que les différentes techniques utilisées.. Si la biométrie est un enjeu important au niveau économique, la recherche, en particulier dans le domaine de la reconnaissance des personnes à partir d'une image 2D de leur visage, offre encore un champ d'investigations très ouvert. Dans le chapitre suivant nous nous intéressons aux réseaux de neurones qui représente le fondement de la technique utilisé notre détecteur.

## III.1 Introduction

Les **réseaux de neurones formels** sont à l'origine d'une tentative de modélisation mathématique du cerveau humain. Les premiers travaux datent de 1943 et sont l'oeuvre de MM. *Mac Culloch et Pitts.* Ils présentent un modèle assez simple pour les neurones et explorent les possibilités de ce modèle. L'idée principale des réseaux de neurones "modernes" est la suivante :

On se donne une unité simple, un ***neurone***, qui est capable de réaliser quelques calculs élémentaires. On relie ensuite entre elles un nombre important de ces unités et on essaye de déterminer la puissance de calcul du *réseau* ainsi obtenu. Il est important de noter que ces neurones manipulent des données **numériques** et non pas symboliques.

Deux visions s'affrontent donc, d'un côté les tenant de la modélisation biologique qui veulent respecter un certain nombre de contraintes liées à la nature du cerveau, de l'autre les tenants de la puissance de calcul qui s'intéressent au modèle en lui-même, sans aucun lien avec la réalité biologique.

Dans ce chapitre, nous dressons d'abord un bref historique concernant les travaux sur les réseaux de neurone artificiels. Nous présentons ensuite la structure des réseaux de neurones et leur fonctionnement. Ensuite les topologies des réseaux de neurones sont présentées. Enfin le mode de paramétrage de ces réseaux ainsi que leurs (ou quelques) domaines d'application et leur limitations concluent notre chapitre.

## III.2 Réseaux de neurones

### III.2 .1 Définition

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 **[10]**.

### III.2 .2 Historique

Les réseaux de neurones artificiels sont construits sur un paradigme biologique. Ainsi les neurologues Warren Sturgis McCulloch et Walter Pitts menèrent les premiers travaux sur les réseaux de neurones à la suite de leur article fondateur intitulé :

« What the frog's eye tells to the frog's brain » [McCulloch et Pitts, 1943]. Ils constituèrent un modèle simplifié de neurone biologique app….elé neurone formel. Ils montrèrent théoriquement

que les neurones formels simples peuvent réaliser des fonctions logiques, arithmétiques et symboliques complexes.

Le premier succès apparu en 1957 quand Frank Rosenblatt [Rosenblatt, 1958] inventa le premier modèle artificiel nommé le Perceptron. C'était le premier système qui pouvait apprendre par expérience, y compris lorsque son instructeur commettait des erreurs.

Puis en 1960 Widrow et Hoff proposèrent un autre modèle, aussi basé sur les travaux de McCulloch et Pitts, nommé l'ADALINE (ADAptive LInear NEuron). Au contraire de l'orientation psycho–physiologique du Perceptron, l'ADALINE est développé dans le contexte de traitement du signal.

En 1969 Marvin Lee Minski et Seymour Papert publièrent un ouvrage nommé « Perceptrons » [Minski et Papert, 1969] et montrèrent les limitations théoriques des modèles de Perceptron et plus particulièrement de l'impossibilité de traiter par ce modèle des problèmes non linéaires. Ils étendirent implicitement ces limitations à tous modèles de réseaux de neurones artificiels.

Seulement quelques chercheurs continuèrent alors leurs efforts. Les plus célèbres furent Teuvo Kohonen, Stephan Grossberg, James Anderson, et Kunihiko Fukushima. Une révolution survient alors dans les années 80 quand on a découvert d'importants résultats théoriques. Les plus célèbres sont la rétro-propagation d'erreur inventée en 1986 et la nouvelle génération de réseau de neurone : le perceptron multicouche proposé par Werbos.

### III . 2 . 3 Le neurone biologique

- **Structure**

Le neurone est une cellule composée d'un corps cellulaire et d'un noyau. Le corps cellulaire se ramifie pour former ce que l'on nomme les dendrites. Celles-ci sont parfois si nombreuses que l'on parle alors d'arborisation dendritique. C'est par les dendrites que l'information est acheminée de l'extérieur vers le soma, corps cellulaire.

L'information traitée par le neurone chemine ensuite le long de l'axone (unique) pour être transmise aux autres neurones. La transmission entre deux neurones n'est pas directe. En fait, il existe un espace intercellulaire de quelques dizaines d'Angstroms ($10^{-9}$ m) entre l'axone du neurone afférent et les dendrites du neurone efférent. La jonction entre deux neurones est appelée la synapse figure (III.1) .

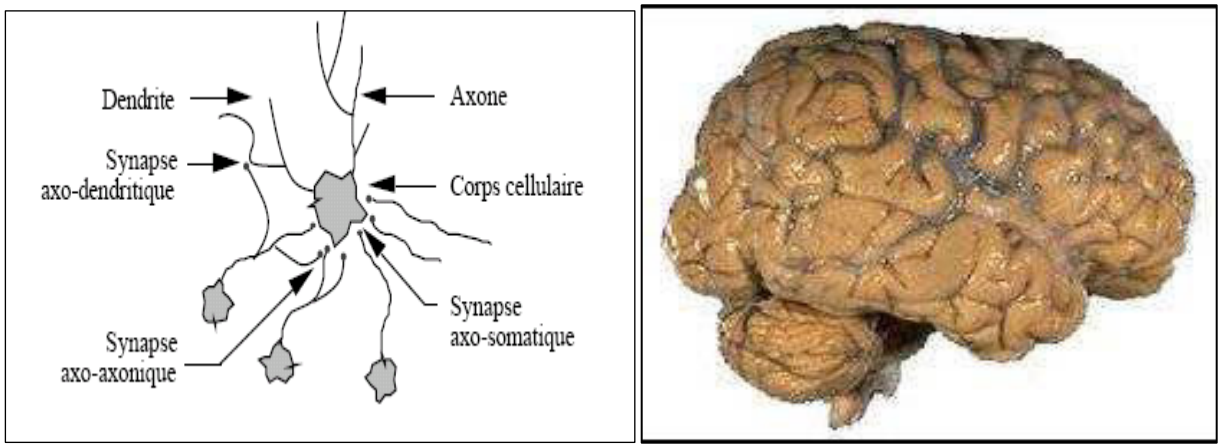



***Figure III.1 :*** *Un neurone biologique et le cerveau humain*

Selon le type du neurone, la longueur de l'axone peut varier de quelques microns à 1,50 mètre. De même les dendrites mesurent de quelques microns à 1,50 mètre. Le nombre de synapses par neurone varie aussi considérablement de plusieurs centaines à une dizaine de milliers.

- **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 micron de longueur.
- **L'axone :** c'est au 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 où il se connecte aux dendrites des autres neurones. Sa taille peut varier entre quelques millimètres à plusieurs mètres.

Chaque neurone est une cellule. Autour du noyau, on trouve le corps cellulaire Celui-ci se prolonge par un axone unique et comporte de nombreuses dendrites qui constituent son organe d'entrée. Chaque neurone est asservi au maintien d'un gradient électrique (potentiel membranaire) d'environ -70 mV entre le milieu extérieur et le soma (intérieur du neurone) [11].

- **L'influx nerveux**

L'influx nerveux ou potentiel d'action figure (III.2) résulte d'une activité électrochimique ; est assimilable à un signal électrique, se propageant dans les neurones de la manière suivante :

- Les dendrites reçoivent l'influx nerveux des autres neurones.
- Le neurone évalue alors l'ensemble de la stimulation qu'il reçoit (c.-à-d. sa dépolarisation par rapport à l'extérieur).

- En fonction de cette stimulation (si la dépolarisation est suffisante > -50mV par exemple), le neurone transmet ou non un signal de type **tout ou rien ;** le long de son axone, selon une fréquence et en fonction du niveau de dépolarisation, nous dirons est ou non excité.
- L'excitation est propagée le long de l'axone jusqu'aux autres neurones ou fibres musculaires qui y sont connectés via les synapses.

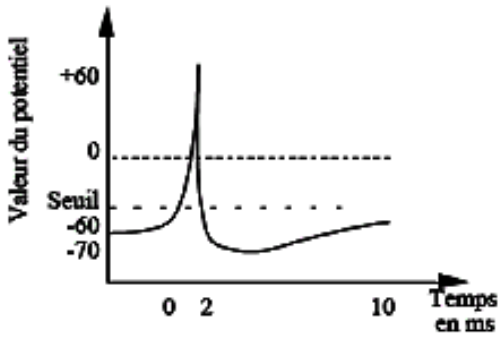



***Figure (III.2) :*** *Un potentiel d'action*

Parties : le corps cellulaire (Somma), l'axone et l'arborisation dendritique

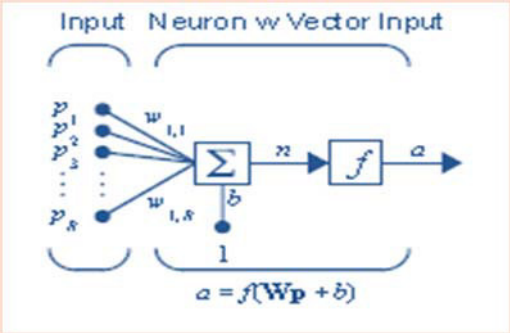
## III.3. Le modèle mathématique

### III.3.1 Le neurone artificiel (formel)

Un "neurone formel" ou simplement "neurone" est une fonction algébrique non linéaire et bornée, dont la valeur dépend de paramètres appelés coefficients ou poids.

Les variables de cette fonction sont habituellement appelées "entrées" du neurone, et la valeur de la fonction est appelée sa "sortie". Un neurone est donc avant tout un opérateur mathématique, dont

on peut calculer la valeur numérique par quelques lignes de logiciel. On a pris l'habitude de représenter graphiquement un neurone comme indiqué sur la Figure (III.3)




**FigureIII.3** Modèle d'un neurone formel

### III.3.2 Fonctionnement

Le neurone de la figure 3.1, réalise une simple somme pondérée de ses entrées, compare une valeur de seuil, et fourni une réponse binaire en sortie. Par exemple, on peut interpréter sa décision comme classe 1 si la valeur de a est +1 et classe 2 si la valeur de a est -1

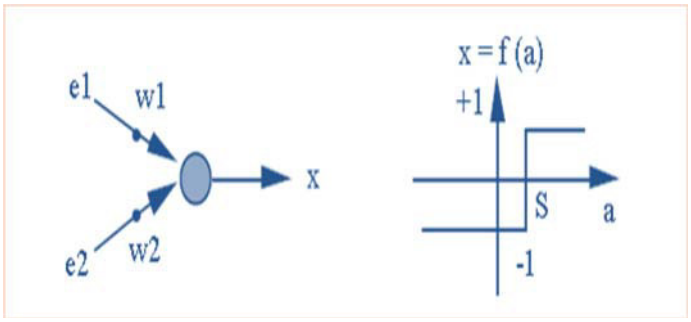
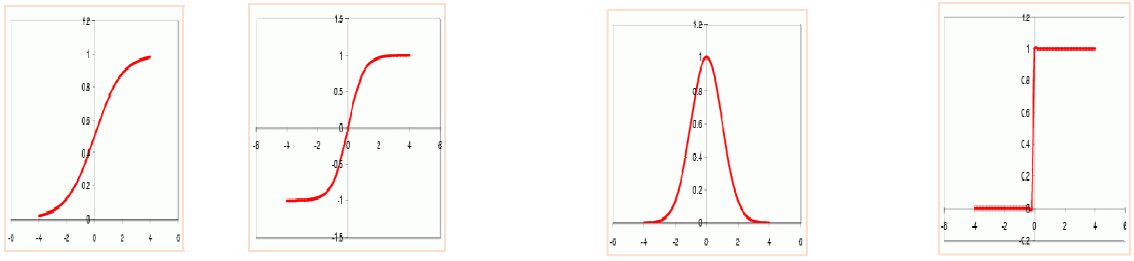



**Figure III.4** Fonctionnement d'un neurone formel

### III.3.3 Les fonction d'activation

Dans sa première version, le neurone formel était implémenté avec une fonction à seuil, mais de nombreuses versions existent. Ainsi le neurone de Mc Culloch et Pitts a été généralisé de différentes manières, en choisissant d'autres fonctions d'activations, comme les fonctions linéaires par morceaux, des sigmoïdes ou des gaussiennes par exemples :



Sigmoïde standard Tangente hyperbolique La fonction Gaussienne La fonction à seuil

**Figure III.5** Quelques types des fonctions d'activations.

## III.4 Architecture du réseau

La **figure III.6** présente une taxonomie possible en terme d'architecture de réseaux .la différence majeure porte sur la possibilité d'avoir des boucles dans le réseaux (cycle ou circuit),ce qui permet au système d'avoir accès (dans une certaine mesure)au passé.par ailleurs, on notera la possibilité d'avoir des « couches » de cellules, c'est-à-dire des groupement de cellules n'ayant aucune interaction, cette notion offre la possibilité de faire transiter le flot de

données dans le réseau de manière séquentielle (entre les couches)et parallèle (au sein d'une même couche).

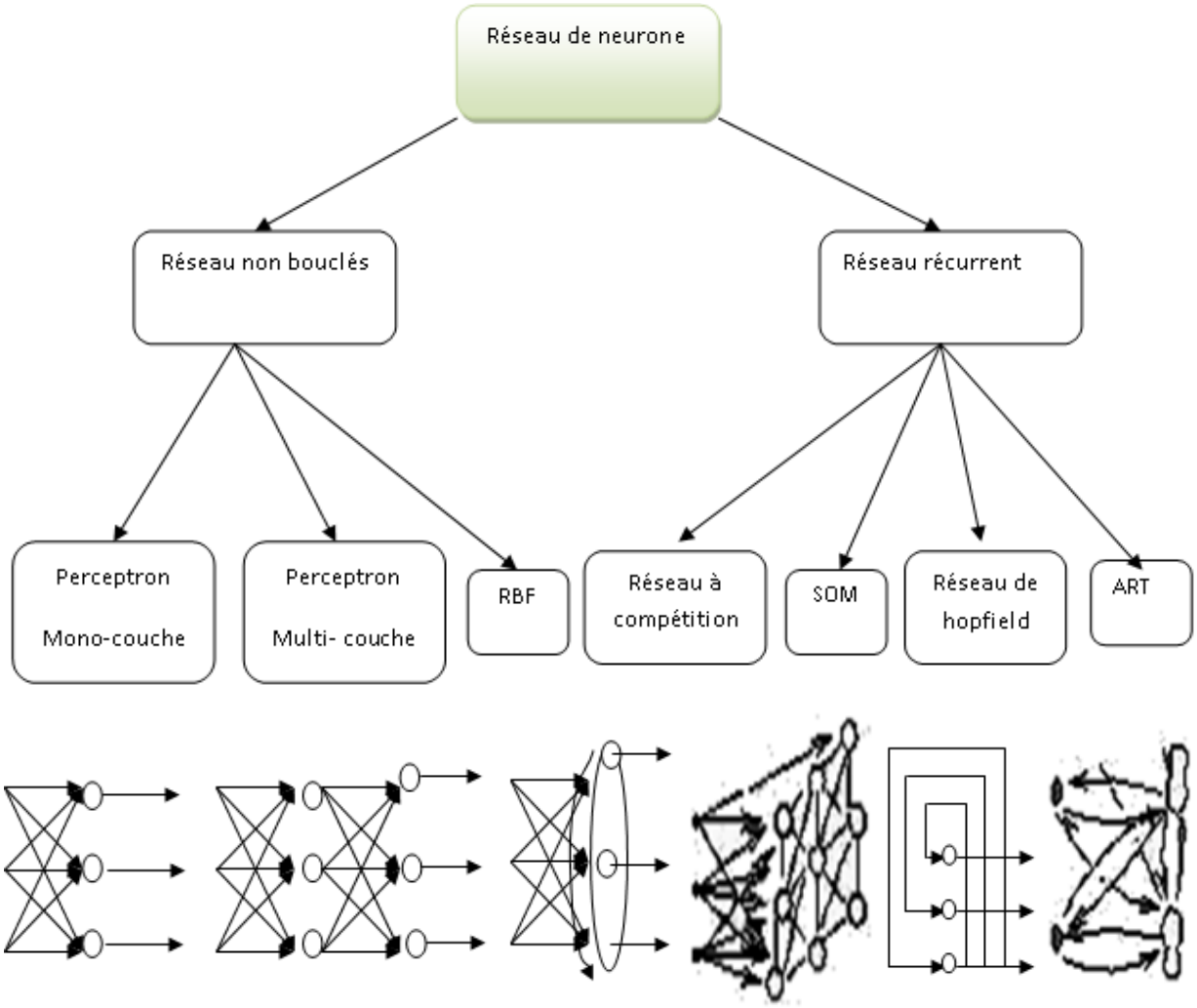



**Figure III.6** Une taxonomie possible

## III.5 Les différents types de réseaux de neurones RNA

Nous donnons dans ce qui suit une classification générale des réseaux de neurones en séparant les réseaux « feed-forward » et les réseaux « feedback ».

### III.5.1 Les réseaux à couches (feed-forward)

#### III.5.1.1 Perceptron

Avant d'aborder le comportement collectif d'un ensemble de neurones, nous allons présenter le Perceptron (un seul neurone) en phase d'utilisation. L'apprentissage ayant été réalisé, les poids sont fixes. Le neurone de la (figure III.7) réalise une simple somme pondérée de ses entrées, compare une valeur de seuil, et fourni une réponse binaire en sortie [12].

Par exemple, on peut interpréter sa décision comme classe 1 si la valeur de x est +1 et classe 2 si la valeur de x est -1.

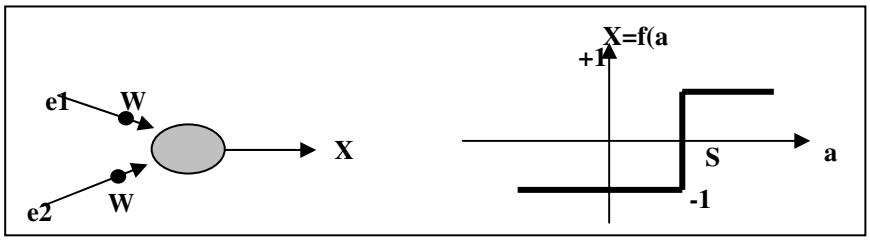



***Figure III.7** : Le Perceptron : structure et comportement*

Les connexions des deux entrées e1 et e2 au neurone sont pondérées par les poids w1 et w2. La valeur de sortie du neurone est notée x. Elle est obtenue après somme pondérée des entrées a et comparaison à une valeur de seuil S.

- **Perceptron monocouche**

Ce sont les réseaux les plus simples. Ils sont utilisables pour des problèmes de classification et d'approximation. Leur avantage est que l'apprentissage du réseau converge vers une solution optimale. Cela est dû au fait que c'est un système linéaire. Leur inconvénient est qu'ils peuvent seulement classifier ou approximer les problèmes linéaires et ne peuvent résoudre un problème non linéaire. Ces réseaux suivent un apprentissage supervisé selon la règle de correction d'erreurs [12].

L'exemple classique pour un système de neurones monocouches est le Perceptron monocouche, inventé par Rosenblatt [Rosenblatt, 1958]. C'est un modèle très simple, basé sur l'orientation physico-physiologique. Il ne dispose que deux couches :

- Une couche d'entrée qui s'appelle la rétine et qui est une aire sensorielle ;
- Une couche de sortie qui donne la réponse correspondante à la simulation présentée à l'entrée

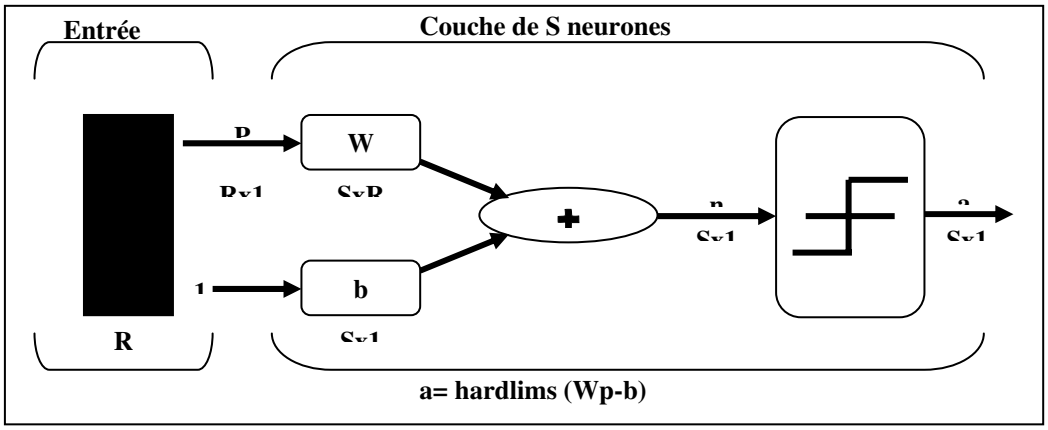



***Figure III.8** : Réseau monocouche*

- **Perceptron Multicouches (PMC)**

Le PMC est un modèle d'une plus grande capacité de calcul. Sa structure est composée d'une couche d'entrée, une couche de sortie, interprétée comme la réponse du réseau et d'une ou plusieurs couches intermédiaires dites « couches cachées ». Un neurone d'une couche inférieure ne peut être relié qu'à des neurones des couches suivantes. Il suit un apprentissage supervisé et utilise une règle d'apprentissage par rétro propagation. En général, les neurones du Perceptron multicouche sont animés par une fonction d'activation non linéaire (au moins dans une des couches). Les choix classique pour cette fonction sont :

- La fonction tangente hyperbolique
- La fonction sigmoïde

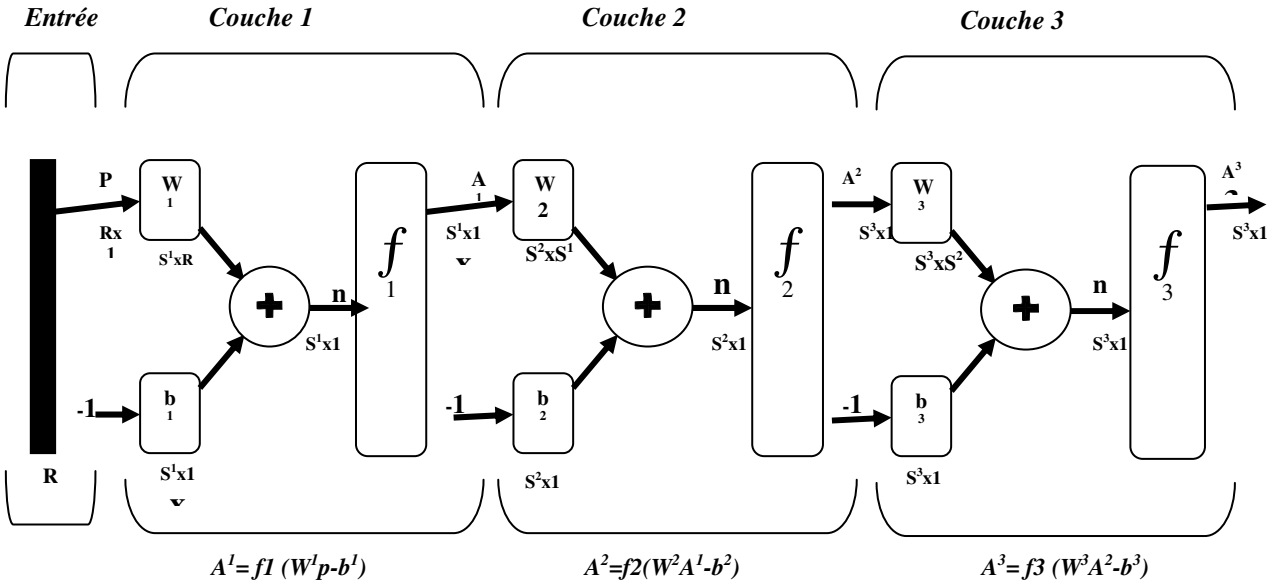



***Figure III.9 :****Réseau multicouches*

- ***Les réseaux à fonction radiale (RBF)***

Le réseau à fonction radiale RBF (Radial Basic Fonction) a la même structure que le PMC, mais la fonction d'activation est une fonction de type Gaussienne. Ce réseau, à cause de son architecture, utilise le plus souvent la règle d'apprentissage de correction d'erreur et la règle par apprentissage compétitif. Il peut avoir un apprentissage qui combine en même temps l'apprentissage supervisé et l'apprentissage non supervisé. Ce réseau obtient des performances comparables ou supérieures à ceux du PMC. De plus leur apprentissage plus rapide et plus simple en fonction des outils de choix pour plusieurs types d'applications, dont la classification et l'approximation des fonctions. Cependant, ce réseau n'est pas assez utilisé que le Perceptron multicouches.

### III.5.2 Les réseaux récurrents (feed-back)

#### III.5.2.1 Carte auto-organisatrice de Kohonen : Le réseau de kohonen Structure

Le réseau de kohonen, aussi appelé SOM (Self Organised Maps): Carte Auto Organiser), est inventé par Teuvo Kohonen en 1982, est généralement constitué d'une grille régulière de neurones en deux dimensions, ayant une topologie, c'est à dire que chaque neurone a un voisinage défini. Ce voisinage est constitué en général des $R$ neurones voisins (à droite, à gauche au dessus et au dessous).

Le réseau est constitué de deux couches seulement. Chaque neurone de la couche d'entrée alimente tous les neurones de la couche de sortie, et tous les neurones de la couche de sortie sont connectés entre eux (voir figure). Un seul neurone de la couche de sortie est actif lorsqu'une entrée est présentée. Le principe général d'un réseau de Kohonen est de faire une classification des entrées et de fournir une sortie qui soit la même pour les entrées proches. (Livre : kazar)

L'activation de chaque neurone de la grille est calculée comme la distance entre les valeurs des neurones d'entrée et des poids associés :

$$D_j = \sqrt{\sum_i (x_i - w_{ij})^2} \qquad \text{(III.1)}$$

Pour une entrée donnée, le neurone ayant la plus faible activation est déclaré vainqueur et correspond à la position de cette entrée dans la grille.

Ce type de réseau suit l'apprentissage non supervisé selon la règle de compétition. La formule de modification des poids des coefficients est :

$$w_{ij}(k) = w_{ij}(k-1) + \eta(x_i - w_{ij}(k-1)) \qquad \text{(III.2)}$$

Où $x_i$ la valeur du neurone $i$ de la couche d'entrée. Avec, seul le neurone excité et ses voisins modifient leur poids.

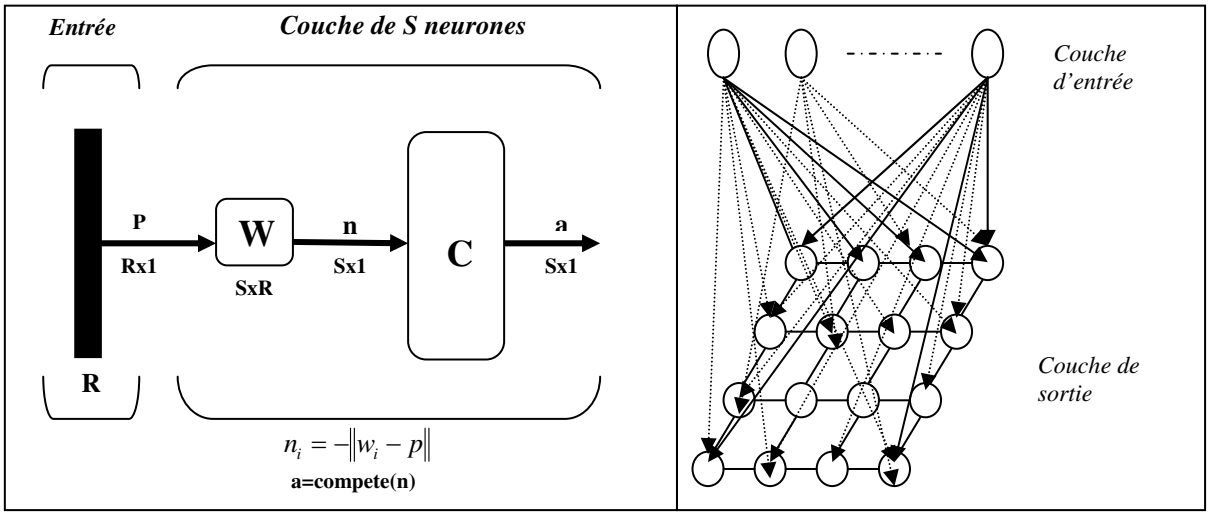



***Figure III.10 :*** *Réseau de Kohonen avec carte carrée .*

### III.5.2.2 Réseau d'Hopfield

Le réseau de neurones d'Hopfield est un modèle de réseau de neurones inventé par le physicien John Hopfield 1982. Un réseau de Hopfield est une mémoire adressable par son contenu.

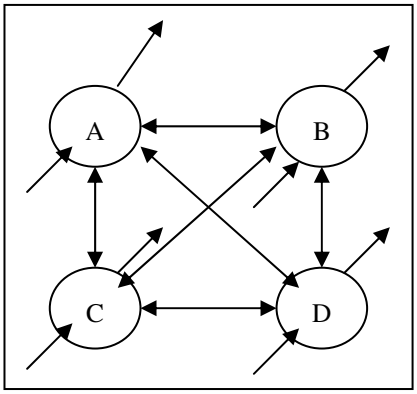



***Figure III.11 :*** *Un réseau de Hopfield à 4 neurones*

Ce modèle de réseau est constitué de N neurones à états binaires (-1, 1 ou 0, 1 suivant les versions) tous interconnectés. L'entrée totale d'un neurone i est donc :

$$I_i = \sum_j w_{i,j} V_j \qquad \text{(III.3)}$$

Où :

- $w_{ij}$ est le poids de la connexion du neurone i à j
- $V_j$ est l'état du neurone j

L'état du réseau peut être caractérisé par un mot de N bits correspondant à l'état de chaque neurone. La mise à jour de l'état des neurones :

- Soit en mode synchrone où tous les neurones sont mis à jour simultanément.
- Soit en mode séquentiel où les neurones sont mis à jour selon un ordre défini.

Le calcul du nouvel état du neurone i se fait ainsi :

$$V_i(t+1) = \begin{cases} 1 & \text{si } \sum_j w_{ij} s_j > 0, \\ -1 & \text{sinon} \end{cases} \qquad \text{(III.4)}$$

Ce type de réseau suit l'apprentissage non supervisé selon la règle de Hebb.

## III.6 L'apprentissage des réseaux de neurones

On appelle « apprentissage » des réseaux de neurones la procédure qui consiste à estimer les paramètres des neurones du réseau, afin que celui-ci remplisse au mieux la tâche qui lui est affectée **[10]**.

L'apprentissage d'un réseau de neurone peut être considéré comme une action de la mise à jour de ses poids des connexions synaptiques, afin de résoudre le problème demandé. L'apprentissage est la caractéristique principale des réseaux de neurones et il peut se faire de différentes manières et selon différentes règles. On peut distinguer deux types d'apprentissage: l'apprentissage supervisé et l'apprentissage non-supervisé. (Voir la figure III.12)

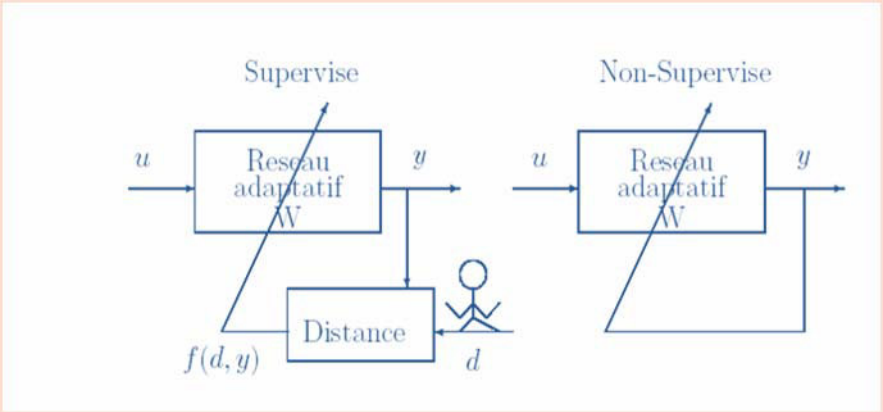



**Figure III.12** Type d'apprentissage en réseau de neurone

Dans le cadre de cette définition, on peut distinguer deux types d'apprentissages : l'apprentissage « supervisé » et l'apprentissage « non supervisé ».

### III.6.1 L'apprentissage supervisé

L'apprentissage supervisé ou l'apprentissage associatif : le réseau adaptatif W compare le résultat y qu'il a calculé, en fonction f (d, y) des entrées u fournies, et la réponse d attendue en sortie. Ainsi le réseau va se modifier jusqu'à ce qu'il trouve la bonne sortie d , c'est-à-dire celle attendue, correspondante à une entrée u donnée. Les différentes réponses sont connues à priori.
On dispose d'une base d'apprentissage qui contient un ensemble d'observation sous forme des couples entrées/sorties associées. Les poids sont modifiés en fonction des sorties désirées **[10]**.

### III.6.2 L'apprentissage non supervisé

L'apprentissage non-supervisé ou auto organisation : l'apprentissage est basé sur des probabilités. Le réseau adaptatif W va se modifier en fonction des régularités statistiques de

l'entrée u et établir des catégories, en attribuant et en optimisant une valeur de qualité, aux catégories reconnues. On ne sait pas à priori si la sortie y est valable ou non. Les entrées sont projetées sur l'espace de réseau **[10]**.

Les deux types ont pour but d'ajuster les poids de connexions entre les neurones, en s'accordant de certaines règles. Plus d'information concernant les règles les plus utilisées dans les différents types des réseaux peuvent être trouvée dansNous citons ci après les règles
Les plus répandues :

### III.6.3 La règle de Hebb

Vue dans le livre « Organisation of Behavior » (1949) [Hebb, 1949]. Cette règle permet de modifier la valeur des poids synaptiques en fonction de l'activité des unités qui les relient. Le but principal est le suivant : si deux unités s'activent en même temps la connexion qui les lie est 14 renforcée (c'est une connexion excitatrice) sinon elle est affaiblie (c'est une connexion inhibitrice) **[10]**.

### III.6.4 La règle de delta

La règle calcule la différence entre la valeur de la sortie et la valeur désirée pour ajuster les poids synaptiques. Elle emploie une fonction d'erreur, nommée aussi le moindre carré moyen, basée sur les différences utilisées pour l'ajustement des poids

### III.6.5 La règle d'apprentissage compétitive

Elle qui ne concerne qu'un seul neurone. On regroupe les données en catégorie. Les neurones similaires vont donc être rangés dans une même classe en se basant sur des corrélations des données et seront représentés par un seul neurone. L'architecture d'un tel réseau possède une couche d'entrée et une couche de compétition. Une forme est présentée à l'entrée du réseau et est projetée sur chacun des neurones de la couche compétitive. Le neurone gagnant est celui qui possède un vecteur de poids le plus proche de la forme présentée à l'entrée. Chaque neurone de sortie est connecté aux neurones de la couche d'entrée et aux autres cellules de sortie (c'est une connexion inhibitrice) ou à elle même (c'est une connexion excitatrice). La sortie dépend alors de la compétition entre les connexions inhibitrices et excitatrices **[10]**.

### III.6.6 La règle de corrélation en cascade

Proposé avec (Fahlman et Lebière, 1990). C'est une technique d'apprentissage qui ajoute progressivement des neurones cachés au réseau jusqu'à ce que l'effet bénéfique de ces nouveaux neurones ne soit plus perceptible. Cette règle suit les deux étapes suivantes :

1. On entraîne le système par un apprentissage classique qui s'effectue premièrement dans un réseau petit sans couche cachée.

2. On entraîne maintenant un petit groupe des neurones supplémentaires qui doit diminuer l'erreur résiduelle du réseau. La règle d'apprentissage utilisée modifie les poids de ces neurones. Le neurone qui réussit le mieux est ensuite retenu, et intégré au réseau. Puis l'étape 1 est relancée, pour permettre au réseau de s'adapter à la nouvelleressource **[10]**.

### III.6.7 La règle de correction d'erreurs

Cette règle peut se caractériser par les étapes suivantes :

- On commence avec des valeurs des poids de connexions qui sont pris au hasard.
- On introduit un vecteur d'entrée de l'ensemble des échantillons pour l'apprentissage.
- Si la sortie ou la réponse n'est pas correcte, on modifie toutes les connexions pour atteindre la bonne réponse **[10]**.

### III.6.8 La règle de rétro-propagation

La règle inventée par Rumelhart, Hinton et Williams en 1986. Elle s'utilise pour ajusteles poids de la couche d'entrée à la couche cachée. Cette règle peut aussi être considérée comme une généralisation de la règle delta pour des fonctions d'activation non linéaire et pour des réseaux multicouches. Les poids dans le réseau de neurones sont au préalable initialisés avec des valeurs aléatoires. On considère ensuite un ensemble de données qui vont servir à l'apprentissage. Chaque échantillon possède ses valeurs cibles qui sont celles que le réseau de neurones doit à terme prédire lorsqu'on lui présente le même échantillon **[10]**.

## III.7 Les limitations d'un réseau de neurones

Les réseaux de neurones peuvent implémenter n'importe quelle fonction non linéaire jusqu'à un certain degré de fiabilité. Ils peuvent implémenter des fonctions dynamiques et posséder n'importe quel nombre d'entrée et de sortie. Nous allons lister quels sont les avantages et les inconvénients des réseaux de neurones artificiels.

Les **avantages** d'utilisation des réseaux de neurones sont :

- Une tolérance à l'incertitude très élevée ;
- Etant une multiple copie d'unité simples (les neurones), ils sont donc facilement extensibles ;
- Une facilité d'application car ne nécessitant pas une compréhension approfondie ;

- Un choix de types, d'architecture et de fonction d'activation de réseaux diverses ;
- Ils possèdent une capacité à la généralisation ;

Les **inconvénients** des réseaux de neurones sont :

- une facilité d'application donnant lieu à de nombreuses implémentations et des choix pas toujours justifié ;
- malgré une solide base théorique, le choix du réseau appartient souvent à l'utilisateur car il n'existe pas de guide approprié et reconnu ;
- La surface d'erreur des réseaux complexes possède beaucoup de sommets (des maximums locaux) et des vallées (des minimums locaux). A cause de leur nature non linéaire ils peuvent être piégé dans un minimum local où les performances des réseaux sont nettement sous optimales.

Pour éviter les minima locaux on doit envisager les solutions suivantes :

1. Modifier le pas d'apprentissage du réseau pour pousser le réseau hors des minima locaux. C'est un paramètre qui règle la taille de la surface d'erreur.
2. Entraîner un même réseau à partir de plusieurs choix initiaux de poids, pour ensuite ne garder que le meilleur d'entre eux.

## III.8 Applications des RNA

L'approche connexionniste semble bien adaptée à des problèmes d'optimisation ou de reconnaissance de formes. Des exemples d'utilisations réussies de réseaux de neurones dans certains domaines, en particulier :

- Dans l'industrie (par exemple le diagnostic de panne ou le contrôle de qualité).
- La télécommunication (par exemple l'élimination du bruit ou l'analyse du signal).
- L'environnement (par exemple la prévision et la modélisation météorologiques ou l'analyse chimique).
- Le marketing est aussi un domaine où les réseaux sont utilisables pour diminuer les tâches d'administration d'énormes bases de données. On appelle cette méthode, le datamining (fouille de donnée).
- La robotique ; par exemple le contrôle de déplacement des robots autonomes.

- La médecine ; par exemple l'identification des arythmies cardiaques, de reconnaissance des lésions pour différentes organes…

## III.9 Conclusion

Nous pouvons dire que les réseaux de neurones sont des modèles de calculs apprenant, généralisant et organisant des données.

Un réseau de neurones artificiels contient un grand nombre d'unités, les neurones, qui communiquent entre eux en s'envoyant des signaux à travers de liens, appelées connexions synaptiques. En général le système de neurone possède trois types des neurones :

- Les neurones d'entrée qui reçoivent les données.
- Les neurones de sortie qui envoient les données par la sortie du système.
- Les neurones cachés, dont les signaux d'entrée et de sortie demeurent dans le système.

Dans le chapitre suivant, nous allons développer la conception de notre système neuronal pour la reconnaissance des visages.

## IV.1 Introduction

L'analyse du contenu des images et la reconnaissance de formes sont des domaines d'applications très utilisés de nos jours et rendus de plus en plus efficaces par la puissance croissante des machines.

La détection de visages, traitée ici, illustre bien les difficultés rencontrées dans ce type d'applications. En effet, toute reconnaissance passe par des critères de reconnaissance. Il faut donc pouvoir définir ce qui est recherché dans l'image. Un visage est quant à lui relativement simple à définir. C'est sa définition qui nous amènera à définir une topologie relativement intuitive pour les réseaux de neurones utilisés.

- **Chaine de traitement de la détection**

Afin de rechercher des critères, il faut permettre à nos données d'être comparables à nos critères, il faut donc au préalable normaliser ces données. C'est ce qui nous amènera tout d'abord à définir un filtrage pré détection. Ensuite, nous pourrons définir la topologie des réseaux. Puis, afin de permettre un apprentissage, il faudra définir ce qui doit être supervisé par l'utilisateur et ce qui peut être automatiquement supervisé par l'application. Seront à cette fin présentées des méthodes et heuristiques allégeant à la fois l'apprentissage et optimisant les résultats. La chaîne de traitement de la **figure IV.1** montre l'enchaînement des étapes de la phase de détection du système de base proposé.


**Figure IV. 1** : Chaine de traitement de la détection

## IV.2 Collecte des données : Balayage et Filtrages

### IV.2.1 Balayage de l'image

L'objectif recherché étant la détection de visages dans une image, il faut d'abord définir une fenêtre qui va parcourir l'image à la recherche d'un visage.

Cette fenêtre doit être de taille fixe pour servir de donnée entrante aux réseaux. C'est ce qui nous amènera à balayer l'image à plusieurs échelles.

Dans l'article [1], les auteurs proposent une fenêtre de taille 20x20 pixels.

Aussi, la détection de visage ne pourra s'opérer que sur des images d'une taille minimale de 20x20 pixels. Le balayage démarrera donc sur une image à sa taille initiale, puis l'image sera successivement réduite à chaque changement d'échelle afin de pouvoir repérer des visages plus ou moins près sur cette image (Figure IV.2).

Les auteurs proposent pour cette phase de déplacer le fenêtre sur chaque pixel de l'image et de prendre 1.2 pour facteur d'échelle. La vitesse de déplacement ainsi que l'échelle peuvent être modifiées mais si par exemple un déplacement de 2 au lieu de 1 divisera par deux le temps d'analyse total, la détection sera moins précise. Il en va de même pour l'échelle. Aussi, ungain en vitesse se traduit immanquablement par une perte en performances de détection.

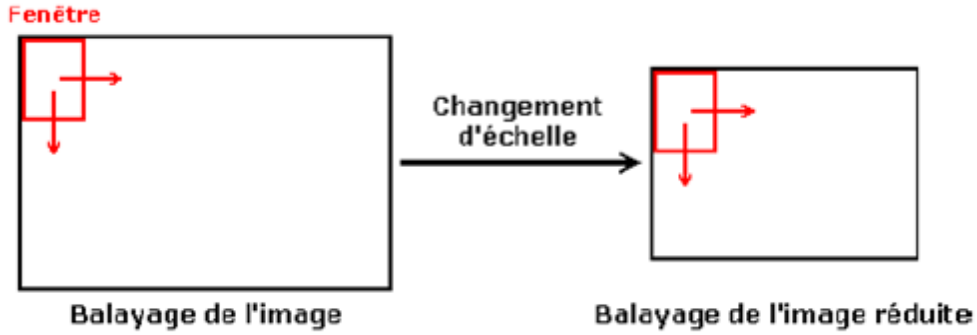



**Figure IV. 2 : P**rocédé de balayage d'une image :

1) balayage à une échelle donnée,

2) réduction d'échelle .

### IV.2.2 Masquage de données

Afin de cibler l'analyse sur l'image, un visage ayant une forme plutôt ovale verticale, les quatre coins de la fenêtre seront ignorés, ainsi qu'une bande de chaque côté, dans la mesure où ces données représenteront principalement un décor, des habits, ou des cheveux, données très variables et inutile pour de la détection de visages. Cette technique amène donc à définir un masque de taille 20x20 pixels pour cacher des données (Figure3).

**Figure IV. 3** : Fenêtre et masque de donnée

### IV.2.3 Passage en niveaux de gris

Dans la recherche de traits de visages, la couleur n'apporte rien et ne peut que gêner la détection, c'est pourquoi l'analyse est effectuée en niveaux de gris. Afin d'obtenir des variations de contraste, l'image, si elle est en couleur, doit donc être passée en niveaux de gris.

### IV.2.4 Normalisation

Maintenant que les données en entrée sont définies, il faut les rendre comparables quelles que soient les images originales. Chaque image peut en effet provenir d'un milieu plus ou moins lumineux ou avec un éclairage et des ombres différentes. Il est donc important, pour qu'un réseau puisse apprendre, que les données soient normalisées et que les variations de contraste soient représentatives de caractéristiques de visages et non de milieux ou d'expositions.

La normalisation des images va s'effectuer en deux temps :

1. Egalisation de la lumière dans l'image,

2. Egalisation par histogramme.


**Figure IV.4 : C**alcul et application du filtre d'égalisation d'intensité

### IV.2.4.1 Egalisation de l'intensité lumineuse de fond

La première étape de ce filtre de normalisation est l'égalisation des intensités rencontrées dans la fenêtre de l'image. Nous nous intéresserons aux pixels non cachés par le masque, l'arrière plan étant hors sujet.

Le principe va être de définir une fonction qui approxime par région de la fenêtre l'intensité globale de cette région et ensuite de la soustraire à la fenêtre. Ceci aura pour effet de conserver les variations locales d'intensité mais de ramener la moyenne globale d'intensité par région à une constante.

Ainsi, l'influence de l'exposition initiale de l'image est très atténuée. Cela permet, sur ce premier critère, une normalisation des images issues de milieux hétérogènes et tend donc à rendre possible une comparaison.

L'image 4 montre l'effet du filtre d'égalisation d'intensité. Cette image est tirée de l'article de référence [18].

### IV.2.4.2 Egalisation de l'histogramme

Bien que rendues plus homogènes, les images à traiter sont maintenant relativement fades. On entend par là que les traits du visages ne ressortent que peu. Or, pour entraîner le réseau à reconnaître des traits, il faut être à même de les mettre en valeur. C'est donc là l'objectif de l'égalisation des contrastes par histogramme.

Ce procédé a pour principe de rendre constante la fréquence des valeurs.

Le mot "valeur" est ici à prendre au sens artistique, par opposition à couleur, qui définit en ce terme les nuances de gris.

Pour ce faire, il faut calculer l'histogramme des valeurs de l'image donnant pour chaque valeur sa fréquence, à savoir le nombre de pixels à cette valeur. Cet histogramme doit avoir en mémoire, pour chaque pixel, sa position dans l'image.

Il faut alors calculer la moyenne définie par :

où le nombre de valeurs est 256 lorsque l'on travail en 8 bits par pixels, chaque valeur

$$m = \frac{Nombre\ total\ de\ pixels}{Nombre\ de\ valeurs}$$

appartenant à l'intervalle [0, 255]. Cette moyenne calculée, elle nous donne le nombre de pixels par valeur qu'il faut avoir pour obtenir un histogramme plat. Ainsi, l'algorithme va itérativement, de 0 à 255, étaler sur les proches voisins le surplus potentiel de pixels, amenant ainsi à diminuer les fréquences supérieures à la moyenne et augmenter celles qui lui sont inférieures. A terme l'histogramme obtenu doit être approximativement plat, donc les écarts à la moyenne pour chaque fréquence proches de 0. La **figure IV.5** montre la transformation de l'histogramme. Une fois l'histogramme égalisé, l'image est reconstruite avec la nouvelle répartition des valeurs. La figure IV.6 tirée de [1] montre l'image ainsi obtenue.

## IV.3 Fonctionnement du Réseau

### IV.3.1 Topologie

Dans toute utilisation de réseaux de neurones, il faut définir une topologie de réseau. Il n'y a aucune règle pour définir cette topologie et c'est souvent par tests successifs qu'une bonne topologie est définie. Cependant, dans le cas présent, nous sommes à même de définir une topologie de base.

Un visage se distingue en effet surtout par des yeux, un nez et une bouche. Aussi, il semble intuitif de définir une topologie segmentant l'image afin de pouvoir repérer de telles caractéristiques. A cette fin, l'article [1] propose une telle topologie :

- 4 unités de 10x10 pixels,
- 16 unités de 5x5 pixels
- 6 unités de 20x5 pixels.

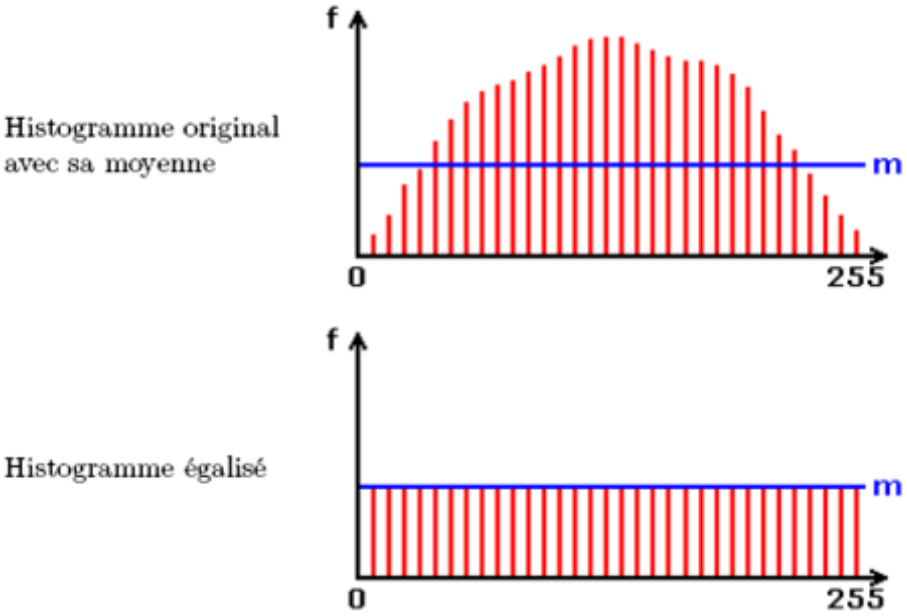



**Figure IV. 5 : P**rincipe de l'égalisation des valeurs dans l'histogramme


**Figure IV.6 :** Application de l'égalisation des valeurs par histogramme

Une unité sera une donc une couche d'entrée de données. Les unités carrées peuvent notamment repérer un œil, un nez, ou un coin de bouche tandis que les unités en bande de 20x5 repère la ligne des yeux ou la bouche. Ainsi, les unités seront entraînées chacune à une tache précise et seront donc spécialisées. C'est en réunissant les réponses de ces unités que l'unité finale pourra dire si la fenêtre contient un visage ou non (Figure IV.7 tirée de [1]).

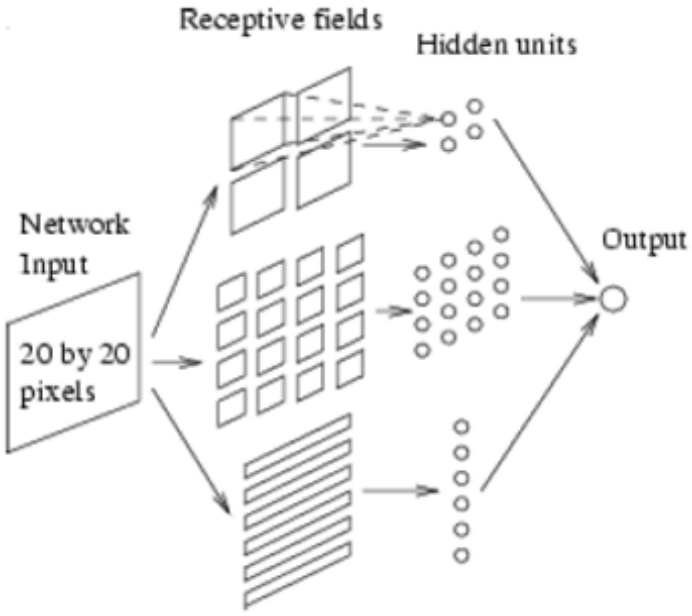



**Figure IV .7 : R**éseau de détection

La topologie de base sera donc d'une unité finale fournissant une réponse binaire (dans l'article [1]) ou probabiliste. On mettra derrière cette unité les couches cachées du réseau, définies plus haut. J'appelle notamment cela une topologie de base car le nombre d'unités, leur taille et leur position restent non empiriques et ne peuvent en conséquence pas à être fermement fixés.

L'article [1], adoptant la topologie citée plus haut, précise de même que le nombre de couches de chaque unité peut être augmenté et les auteurs utilisent pour leurs expériences des doubles et triples couches.

### IV.3.2 Mise en place

Les réseaux de neurones les plus répandus et les plus simples à la fois restent les perceptrons multi-couches (PMC) qui consistent en une succession de couches, interconnectées totalement ou partiellement. Il est évident qu'un tel réseau brut ne peut convenir à la topologie citée plus haut. Cependant, une propriété intéressante d'un réseau de neurones est que tout sous réseau est un réseau. Ainsi, tout réseau est un méta réseau. De là, on peut très simplement définir un réseau respectant la topologie citée plus haut à partir de plusieurs PMC. L'algorithme d'apprentissage total reviendra à transmettre à tous les PMC, traitant chacun une unité, le résultat attendu. Si l'exemple à apprendre est un visage, la première étape transmet aux PMC des yeux, de la bouche et du nez, que l'on attend une réponse positive. De là, chaque PMC applique son algorithme d'apprentissage. La **figure IV 8** illustre de manière simplifiée cette mise en place.

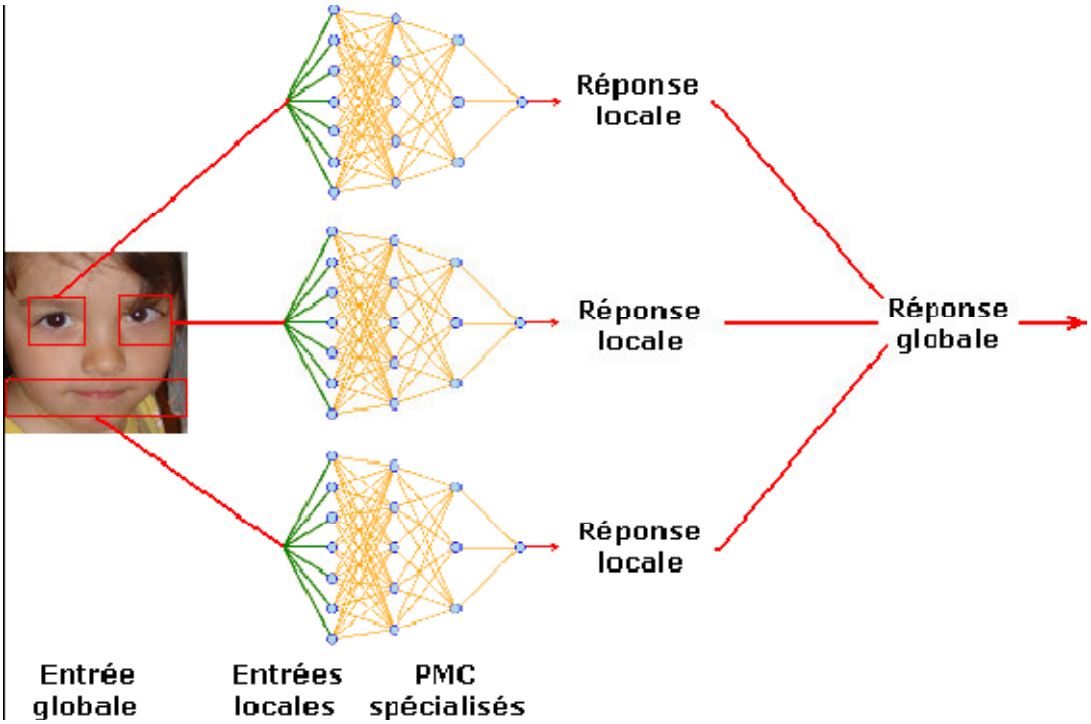



**Figure IV .8 :**Mise en place d'un réseau de PMC

## IV.4 Apprentissage

L'apprentissage d'un réseau de neurones étant supervisé, il faut pouvoir définir des exemples d'apprentissage, ainsi que des contre-exemples

### IV.4.1 Exemples

En premier lieu, il faut posséder une certaine base d'images de visages. A partir de ces images, il va être possible d e générer d'autres exemples, proches des originaux mais qui apporteront de la robustesse aux réseaux. En effet, si nous nous intéressons à la détection de visages verticaux, la verticalité et le centrage ne sont jamais parfaits. On peut donc créer à partir d'un visage exemple des images de ces visages tournées de quelques degrés de chaque côté, translatées de quelques pixels, ou zoomés quelque peu (**Figure IV.9** ). On peut également faire une symétrie planaire, à savoir l'image du visage dans un miroir. Ainsi, on extrait de chaque exemple plusieurs exemples ayant de nouvelles caractéristiques malgré leur proximité au visage original.

**Figure IV .9** : Visages des auteurs retournés, pivotés, zoomés légèrement [18]

Cependant, il faut rester prudent sur ce type d'opération. En effet, autoriser une trop grande rotation ou translation aura pour effet de rendre la détection moins précise. Par exemple, en ne translatant que d'un pixel, le filtre tend à être insensible à un décalage de la fenêtre de un pixel dans l'image originale. Cette insensibilité est raisonnable mais on se retrouve une fois encore face à la dualité rapidité qualité. Cette technique apportant plus d'exemples amène aussi à plus l'imprécision si on la surexploite.

### IV.4.2 Contre-exemples

S'il est relativement simple de réunir des exemples de visages, il n'en va pas de même pour les "non visages". En effet, un visage est quelque chose de définit. Or ce qui n'est pas un visage ne se définit pas. Tout ce qui n'en est pas un, à savoir une infinité de choses, est un contre exemple potentiel. Dans l'idéal, un réseau doit pouvoir apprendre ce qui est et ce qui n'est pas l'objet à détecter. Pour approcher la diversité des contre-exemples de visages, il est cependant possible de proposer des solutions efficaces apportant des images suffisamment hétérogènes.

Tout d'abord, toute image, naturelle ou générée, ne contenant pas de visage, contient des contre-exemples. Afin que le réseau apprenne, on peut lui faire analyser ce type d'image et toute détection de visage sera une erreur, donc un contre-exemple sur lequel il devra apprendre. Ainsi, on peut définir de manière relativement automatisée une procédure d'extraction de contre exemples correcteurs. Cette méthode est d'autant plus intéressante qu'il peut exister dans des images des configurations possédant des points communs avec des traits de visages. Des taches sombres environs aux emplacements des yeux, du nez et de la bouche peuvent induire un réseau en erreur. Ainsi, avec ce type de contre-exemple, il apprendra à être plus exigeant sur la précision dans ces zones. La **figure IV.10**, illustre ce procédé.

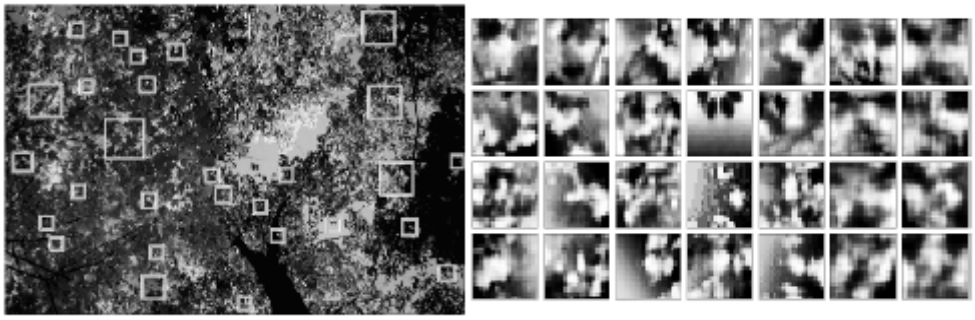


**Figure IV .10 :** Détection dans une image ne contenant pas de visages et visualisation des fenêtres trouvées.

Ces fenêtres ont pour certaines en effet des caractéristiques assez flagrantes de visage au niveau des variations de valeurs.

## IV.5 Sélection pertinente de solutions

L'imprécision et l'insensibilité à quelques pixels près amène très souvent de multiples détections d'un même visage dans un voisinage proche. Il faudrait pouvoir, parmi ces multiples propositions, en sélectionner ou en agréger une. De même, considérant cette propriété, une détection isolée peut être considérée comme une erreur.

**Figure IV.11 :** Détections multiples de visages et erreurs isolée

Cette observation amène les auteurs de à proposer deux méthodes de sélection :

- l'agrégation de multiples détections,
- la sélection selon les avis de plusieurs réseaux de neurones.

### IV.5.1 Agrégation de multiples détections

La **figure IV.11** extraite de [19] montre la détection multiples de visages ainsi que des détections erronées isolées.

Les auteurs proposent donc une heuristique éliminant une bonne partie des détections erronées. Il s'agit de compter, dans un voisinage restreint, le nombre de détections d'échelles proches. Si ce nombre est supérieur à un certain seuil définit au préalable, la zone est considérée comme un visage. Si par contre ce nombre est inférieur à ce seuil, alors la détection est considérée comme erronée. Afin de définir la détection conservée, le centre de chaque fenêtre est calculé. Tous ces centres vont prendre, dans une image appelée pyramide, la valeur du nombre de centre de détections qu'ils approchent dans un voisinage définit par un seuil. Cela va amener au calcul de centroïdes, centres unissant les points du voisinage approchant un minimum, définit par un seuil, de détections. Ces centres, jusqu'ici, peuvent définir des premières détections.

Maintenant, une deuxième heuristique est ensuite proposée, plus arbitraire à mon sens, dans laquelle il s'agit d'éliminer les détections qui débordent sur une autre détection validée, à savoir considérée comme correcte. Appliquée au calcul des centroïdes, cette heuristique va donc traiter sé-quentiellement les centroïdes en commençant par celui ayant le plus de détections à son actif. Ensuite, tout centroïde, représentant donc moins de détections, qui amène à une détection débordant sur une précédente, est supprimé. Ainsi, à la fin, il ne reste que des centroïdes ayant un certain poids de détection et ne concurrençant aucun autre centroïde. Cette heuristique élimine dans la plupart des cas des détections fausses. Cependant, dans une recherche de visage par exemple dans un lieu où beaucoup de gens se croisent, les personnes situées aux premiers plans seront repérées là où ceux plus en retraits seront élagués. Ce type d'heuristique doit donc être utilisée avec prudence selon l'image analysée et l'objectif recherché pour cette détection. Si en effet, il s'agit de rechercher un visage dans une foule en combinant détection et reconnaissance de visages, cette heuristique présente des risques certains (Figure IV.12 ).

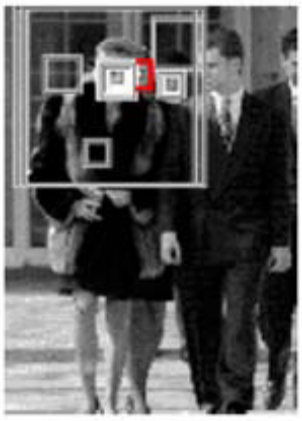


**Figure IV.12 :** Elimination à tort (détection marquée en rouge) par l'heuristique éliminant les détection débordant sur des visages détectés

### IV.5.2 Sélection selon plusieurs avis de réseaux

Une autre approche pour réduire le nombre d'erreurs consiste à entraîner plusieurs réseaux et ensuite à choisir en fonction de leurs réponses, celles qui doivent être conservées ou éliminées.

Afin de rendre efficace ce type de méthode, on considère que les poids initiaux des réseaux sont distribués aléatoirement. De même, la sélection et l'ordre des exemples pour l'apprentissage sont propres à chaque réseau. Ainsi on élimine la possibilité de duplication parallèle, même partielle, des réseaux. Si l'on utilise un algorithme d'apprentissage de type rétro propagation

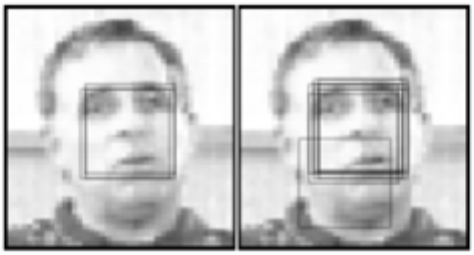


**Figure IV.13** : Détections de deux defferents reseau

**Figure IV.14** : Sélection effectuée par l'opérateur « FT » du gradient, il n'y a pas de facteur aléatoire, il faut donc impérativement un maximum de différences initiales et

d'apprentissage. Dans le cas d'un algorithme génétique, une base commune aura moins de conséquences car les croisements et mutations seront basés sur des facteurs aléatoires.

Quelle que soit la méthode employée, on peut donc arriver à des réseaux faisant des erreurs différentes, et si l'apprentissage est suffisant, leur convergence d'avis donnera en général les détections justes (Figure IV.13et IV.14). Les détections sélectionnées seront celles qui coïncident précisément en terme de position et d'échelle, l'équivalent d'un "ET" logique. Ainsi, il est peu probable que deux réseaux s'accordent sur une erreur, compte tenu de leurs différences initiales et d'apprentissage. Cependant, si un réseau ne détecte pas bien un visage, même si l'autre l'a bien fait, cette détection ne sera pas sélectionnée. Toutefois, ces cas sont très rares.

Cette technique peut être appliquée avec n réseaux et la sélection des visages peut aussi se faire par ou "OU" logique. Bien sur, davantage de réseaux peut entraîner plus de désaccords. De même un "OU" n'élimine pas les fautes mais les réuni. Cependant, couplé avec la technique d'agrégation, les multiples détections étant de fait plus nombreuses encore (somme de plusieurs réseaux), cela permet d'être plus exigeant sur le seuil minimal des détections multiples et donc d'éliminer avec moins d'erreurs les mauvaises détections. Il y a donc plusieurs approches à utiliser et à combiner, chacune possédant des propriétés intéressantes.

## IV.6 Extension à des visages non verticaux

J usqu'ici, les visages détectés sont supposés être droits. La technique employée ne permet pas de détecter un visage écarté de plus de quelques degrés de l'axe vertical. Même si cette configuration verticale est la plus courante, il n'est pas envisageable dans le cas réel d'en faire une hypothèse aussi forte. Une première idée pourrait être de conserver le système en place, et, de même que le processus est répété sur l'image réduite, il pourrait être répété sur l'image pivotée. Cependant, en admettant par exemple que notre détection soit insensible à 10r degrés près, ce type de technique demanderait au moins 18 répétitions du processus (donc une rotation de 20r à chaque fois), déjà lourd en soi. Cette technique, trop lourde en calcul, n'est donc pas envisageable.

L'article [19] apporte à ce problème une solution bien plus rapide. Dans la même idée que la chaîne de traitement vu précédemment, il va s'agir d'introduire une étape pré détection supplémentaire.

### IV.6.1 Principe

L'idée proposée va être d'utiliser un nouveau réseau de neurones, déterminant l'angle d'un visage par rapport à l'axe vertical. Ce réseau suppose que l'image présentée est un visage, il n'y a pas à ce niveau d'attente sur la détection. En effet, il sera entraîné à reconnaître l'angle d'un visage par rapport à l'axe vertical. Si la fenêtre n'en contient pas, il renverra une valeur angulaire sans importance puisque la détection ne retiendra pas l'image. La puissance de cette méthode va être que pour toute fenêtre, l'étape de rotation n'est appliquée qu'une seule fois.

### IV.6.2 Fonctionnement

Les étapes de mise en oeuvre de la rotation d'image ont une structure assez similaire à ce qui a été vu auparavant. Chaque fenêtre passe à travers l'égalisation par histogramme. Ensuite l'image résultante est présentée à un réseau de neurones qui renvoie le degré d'écart à l'axe vertical. Ensuite, la fenêtre initiale est pivotée, puis la chaîne précédente s'applique, à savoir égalisations pré détection puis détection.

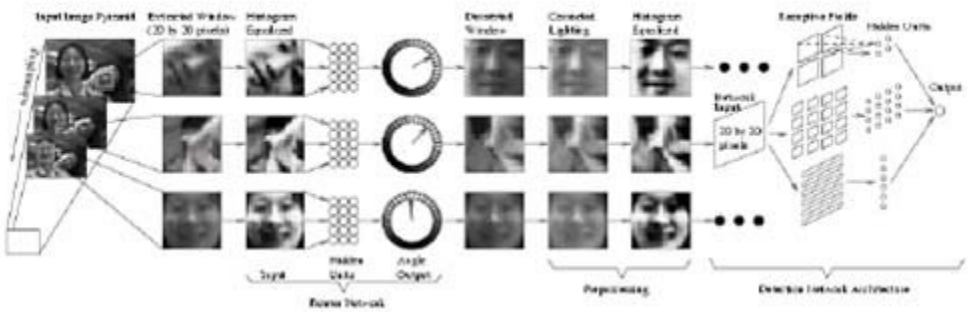


**Figure IV.15 :** Chaine de traitement avec rotation de l'image

### IV.6.3 Apprentissage et Topologie en sortie

L'apprentissage du réseau de rotation va être différent de l'apprentissage utilisé pour la détection. En effet, le réseau ne doit pas renvoyer une valeur booléenne mais permettre d'obtenir un angle de rotation. Il n'y aura donc dans cet apprentissage aucun contre-exemple. La sortie du réseau ne va pas être une valeur mais un vecteur de 36 valeurs. Lors de l'apprentissage, ces valeurs, supervisées, seront :

$$\cos(\theta - i \times 10)$$

avec

- θ l'angle connu du visage sur l'exemple d'apprentissage,
- i une valeur comprise entre 0 et 35.

De même que pour la phase de détection, les images d'apprentissage sont légèrement tournés, translatés et réduites et produisent d'une part une batterie d'exemple et d'autre part par voie de conséquence une insensibilité à de légères rotations, translations ou réductions.

### IV.6.4 Topologie générale

Dans cette partie, il n'est plus possible de définir une topologie aussi précise que lors de la détection. Le réseau n'a à priori aucune information sur la disposition des formes à rechercher comme les yeux, le nez ou la bouche pour la détection. Ainsi, dans ce genre de cas, le perceptron multi-couches à connection totale entre les couches est souvent employé. Les auteurs de proposent une première couche en entrée de 400 unités, soient autant de pixels que la fenêtre donne en entrée, puis une couche cachée de 15 unités, et enfin la couche de sortie de 36 unités (Figure IV.16). Ce type de topologie relève plus de l'expérience de tests réalisés que d'une étude formelle. Plus le réseau est gros, plus il est coûteux en calculs, mais souvent plus il est précis.

Il faut donc trouver, et souvent par le test, un juste milieu. Il en va de même pour les fonctions d'activations. Ici, ils ont choisi une tangente hyperbolique.

L'algorithme d'apprentissage est quant à lui très souvent le célèbre algorithme de rétro propagation du gradient comme c'est le cas dans leur méthode. Cependant, les algorithmes de type génétique, d'une approche totalement différente, non formelle, donnent souvent des résultats comparables voire meilleurs selon les cas.

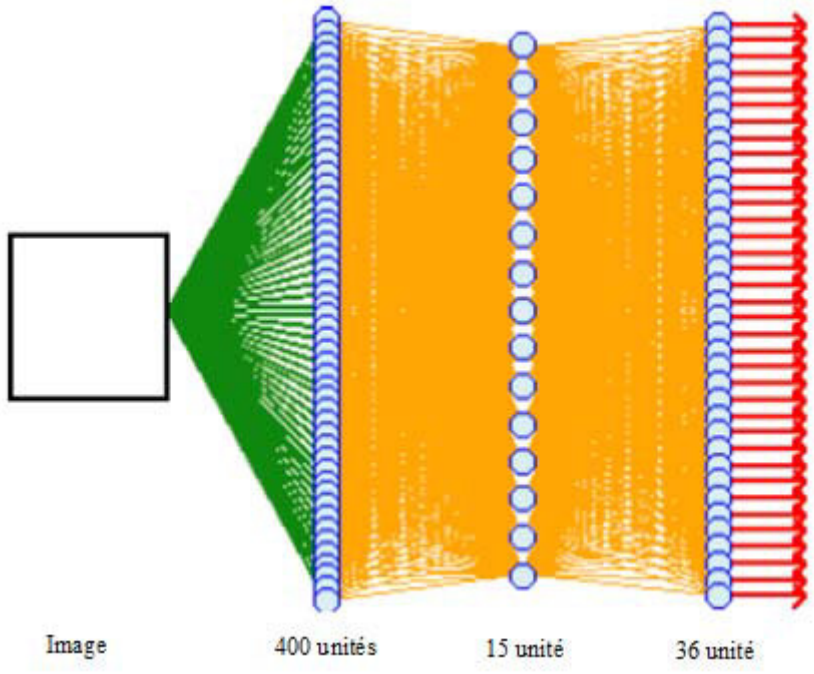



**Figure.16:** Réseau de rotation

## IV.7 Détection grossière des yeux, du nez et de la bouche

### IV.7.1 Introduction

Il s'agit de détecter le centre des yeux, le bout du nez et les deux coins de la bouche. Pour cette étape l'architecture proposée est un réseau de neurones perceptron multicouches entièrement connecté.

### IV.7.2 Base de données utilisée

Je à notre disposition une base de 320 images de visages différents, hommes et femmes ayant diverses particularités (barbe, moustache, frange recouvrant une partie de l'oeil…). Tous les visages ont été photographiés de face et avec une expression neutre. Pour normaliser les visages, une partie de la photo est sélectionné de sorte que la hauteur et la largeur de la nouvelle photographie soit de 3 fois la distance interoculaire (DI)

et que les yeux soient toujours placés au même endroit[19] :

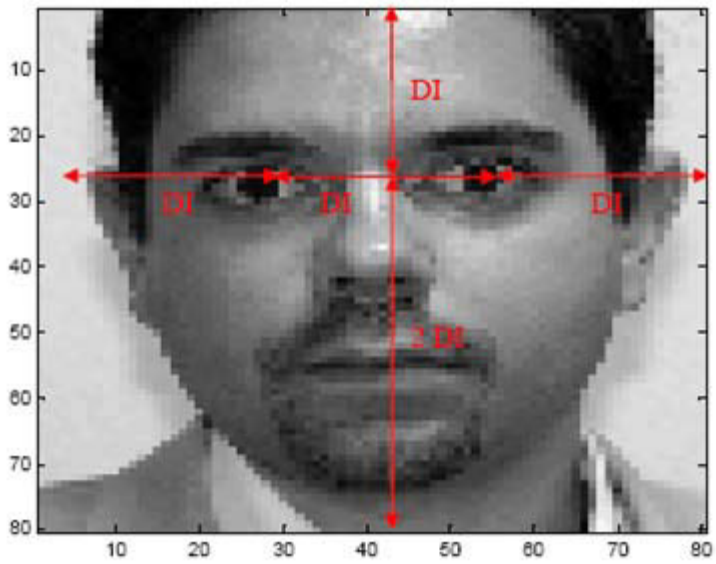



**Figure IV.17:** Normalisation de visages

Puis, pour rendre notre réseau de neurones moins sensible aux translations et variations d'échelle, à partir de chaque visage sont créées des images décalées de 10% de la distance interoculaire dans les 8 directions ainsi que deux images dont les cotés sont agrandis ou rétrécis de 20% de la distance interoculaire (effet de zoom arrière et zoom avant illustré dans la figure 9). Toutes les images sont ensuite sous-échantillonnées pour

obtenir des images de taille 30x30 ou 20x20 pixels.

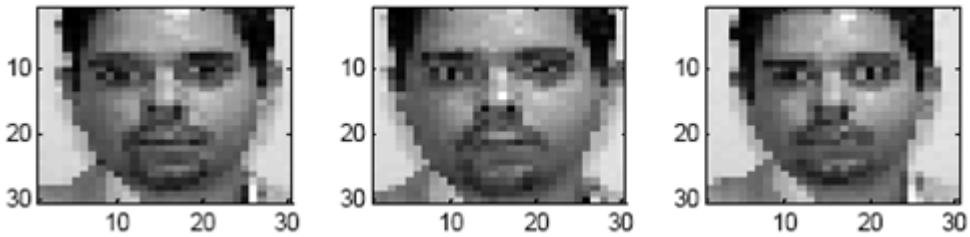



**Figure IV.18:** Exemple d'images utilisées : visage 30x30 et 2 images créées par zoom avant et zoom arrière[19].

### IV.7.3 Mise en oeuvre expérimentale.

Plusieurs configurations différentes ont été testées pour paramétrer ce réseau de neurones. Pour la couche d'entrées, plusieurs cas ont été traités : utilisation directe des niveaux de gris d'images 20x20 ou 30x30 ou analyse préalable en composantes principales sur des images 20x20, 30x30 ou 60x60. En sortie, deux cas ont été envisagés : utilisation des coordonnées des points cibles (donc comme l'on recherche cinq cibles cela équivaut à mettre en sortie un tableau de 10 chiffres) ou mettre en sortie une image de taille 20x20 représentant les positions des différentes cibles.

A chaque fois, des tests ont été réalisés pour trouver le nombre nécessaire de cellules cachées et d'itérations permettant d'obtenir des résultats optimaux. 80% des images (environ 2400 images) ont été utilisées pour l'apprentissage du réseau de neurones et 20% (600 images) pour la cross-validation et le calcul de l'erreur en test. Les images de la base d'apprentissage proviennent de visages que l'on ne retrouve pas dans la base de test. A chaque fois 3 apprentissages par rétro-propagation du gradient de l'erreur ont été réalisés avec le même set de paramètres mais en changeant les visages utilisés pour la base d'apprentissage (qui sont choisis aléatoirement à chaque réalisation), ceci afin de diminuer l'influence du choix de la base d'apprentissage et de lisser la variance des résultats obtenus.

L'erreur est calculée à partir de la distance euclidienne entre la position d'une caractéristique donnée par le réseau de neurones et la position réelle de cette caractéristique. Cette distance est comparée à la distance interoculaire. Ainsi les erreurs de position données dans la suite de ce rapport sont exprimées en pourcentage de la distance interoculaire[19].

### IV.7.4 Sortie du réseau de neurones

Dans un premier temps une carte de caractéristique de même taille que l'image en entrée est utilisée. Cette carte a initialement pour valeur +1 au niveau des caractéristiques (centre des yeux, nez et coins de la bouche) et -1 ailleurs puis elle est lissée par convolution par un filtre gaussien 3x3. Dans le cas d'image 20x20 cela donne

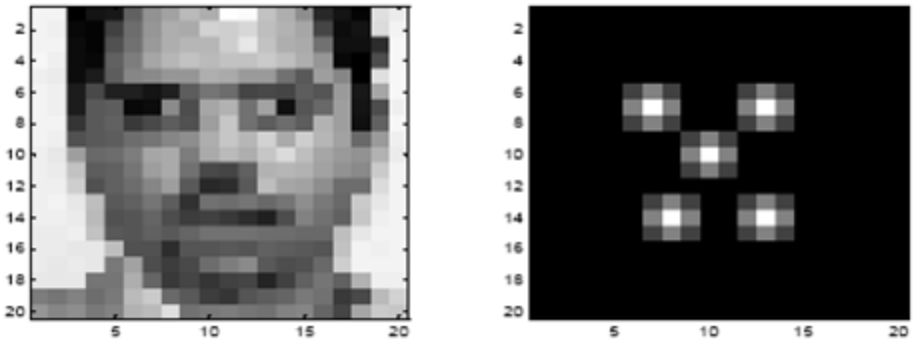



**Figure IV.19 :** Image 20x20 et sa carte de caractéristique associé.

Après l'entraînement du réseau de neurones, pour obtenir la position des caractéristiques, on cherche les maxima locaux dans la sortie donnée par le réseau. Le réseau de neurones engendré de taille 400xNbHCellx400 est comparé au réseau de taille 400xNbHCellx10 ou cette fois la sortie du réseau de neurones correspond à un tableau de 10 chiffres représentant les positions (x,y) des 5 caractéristiques. NbH Cell représente le nombre variable de neurones dans la couche cachée[19].

## IV.8 Conclusion

Cette application des réseaux de neurones montre la simplicité avec laquelle, exemples en main, cet outil est utilisable pour obtenir rapidement un premier résultat. Cependant, ce sont la précision de la détection et le taux d'erreurs qui sont sujets à la plupart des travaux. L'efficacité et la robustesse d'un système se jugeront surtout sur les heuristiques et les méthodes employées pour l'optimiser. Ces méthodes doivent d'une part minimiser le taux de mauvaises détections, et d'autre part maximiser le score de bonnes détections. Ces deux scores ne sont pas totalement complémentaires, nous avons en effet discuté des dangers de l'heuristique éliminant toute détection en chevauchant une autre plus probable. Ainsi, l'erreur se définit à la fois en fonction du nombre de mauvaises détections, mais aussi du nombre de détections omises. C'est ce balancement entre ces deux facteurs qui rend difficile l'optimisation. Les méthodes sont en général efficaces pour l'un mais présentent des effets de bords pour l'autre. C'est ainsi que traiter l'ensemble des cas peut complexifier exponentiellement le traitement pour une amélioration parfois logarithmique.

De plus, si l'on considère tous les angles, en trois dimensions, sous lesquels peuvent être présenté un visage, la reconnaissance devient beaucoup moins triviale. Pour traiter la reconnaissance de visages tournés en dehors du plan de l'image, comme des profils, plusieurs méthodes peuvent être envisagées.

L'utilisation de la symétrie et de la forme du visage peut amener à déduire depuis un visage vu de côté sa forme frontale. Une autre méthode, analogue à celles vues ici, serait d'entraîner séparément des réseaux à reconnaître l'angle de présentation, comme profil gauche, semi profil gauche, face, semi profil droit, profil droit. Cependant, ces angles, bien que les plus courants, ne sont pas les seuls.

## V.1 Introduction

Pour la réalisation d'un système de détection des régions d'intérêt de visage d'une façon automatique, l'extraction de caractéristiques est effectué d'une façon manuelle et par le réseau de neurones. Nous essayons dans ce chapitre de comparer les résultats obtenus par les deux experts la perception humaine et RBF, ensuite un calcul d'erreur quadratique est réalisé pour estimer les performances du réseau de neurone étudié. Dans ce chapitre nous tentons de présenter les différentes procédures constituants le système de détection ainsi que son application à la base de données universelle XM2VTS très utilisée dans les systèmes de reconnaissances. Pour cette raison nous jugeons bon de la présenter en premier.

## V.2 La base de données XM2VTS

Cette base de données XM2VTS a été prolongée de M2VTS (Multi Modal Vérification for Teleservices and Security applications), par le centre CVSSP (Centre for Vision, Speech and Signal Processing), de (l'université de Surrey ,en grande Bretagne, dans le cadre du projet européen ACTS qui traite le contrôle d'accès par une vérification multimodale d'identité , afin de comparer les différentes méthodes de vérification d'identité **.**

La base de données multimodale XM2VTS offre des enregistrements synchronisés des Photos de visages prises de face et de profil et des paroles de 295 personnes des deux sexes Hommes et femmes de différents âges. Pour chaque personne huit prises ont été effectuées en quatre sessions distribuées pendant cinq mois afin de prendre en compte les changements d'apparence des personnes selon plusieurs facteurs (lunettes, barbe, coupe de cheveux, pose..), et chaque session est composée de deux enregistrement, une pour les séquences de parole et l'autre pour les séquences vidéo de la tête. Les vidéos et photos sont en couleur de haute résolution (format ppm), la taille est de 256 x 256 pixels pour les images et de très bonne qualité codé sur 24 bits dans l'espace RGB. Cela permet de travailler en niveaux de gris ou en couleur.

Le choix principal de XM2VTS est sa taille grande, avec 295 personnes et 2360 images en total et sa popularité puisqu'elle est devenue une norme dans la communauté biométrique audio et visuelle de vérification multimodale d'identité.

Nous ne nous intéresserons évidemment, dans le cadre de cette mémoire, qu'aux images prises de face pour le processus de l'authentification de visage.

### V.2.1 Le protocole de XM2VTS ou "protocole de Lausanne

L'existence d'une base de données pour la vérification d'identité nécessite un protocole rigoureux qui permet la comparaison entre les algorithmes de vérification.

Donc, ce protocole de Lausanne est lié directement à la vérification d'identité. Sont principe est de diviser la base de données en deux classes, 200 personnes pour les clients, et 95 pour les imposteurs. Il partage la base de données en trois ensembles : l'ensemble d'apprentissage, l'ensemble d'évaluation, et l'ensemble de test**.**

- **L'ensemble d'apprentissage :** est l'ensemble de référence. Il contient l'information concernant les personnes connues du système (seulement les clients).
- **L'ensemble d'évaluation :** permet de fixer les paramètres du système de reconnaissance de visage.
- **L'ensemble de test :** permet de tester le système en lui présentant des images de personnes lui étant totalement inconnues.

Les imposteurs de l'ensemble de test ne doivent pas être connus du système, ce qui signifie qu'ils ne seront utilisés que pendant la toute dernière phase de test, lorsque le système est supposé fonctionnel et correctement paramétré.

La figure suivante représente des échantillons de notre base de données XM2VTS, de plusieurs personnes, et chaque personne a différents poses.

**Figure V.1** : Exemples des images de la base de données XM2VTS

### V.2.2 Les configuration de notre base de données

Il existe deux configurations différentes, la configuration I et la configuration II. Nous n'utiliserons la configuration I dans cette mémoire. Dans la configuration I, pour la formation de l'ensemble d'apprentissage trois images par client sont employées afin de créer les caractéristiques ou modèles clients. L'ensemble d'évaluation est constitué de trois autres images par clients, ils sont utilisés essentiellement pour fixer les paramètres de l'algorithme de reconnaissance ou de vérification des visages. L'ensemble de test est formé par les deux autres images restantes.

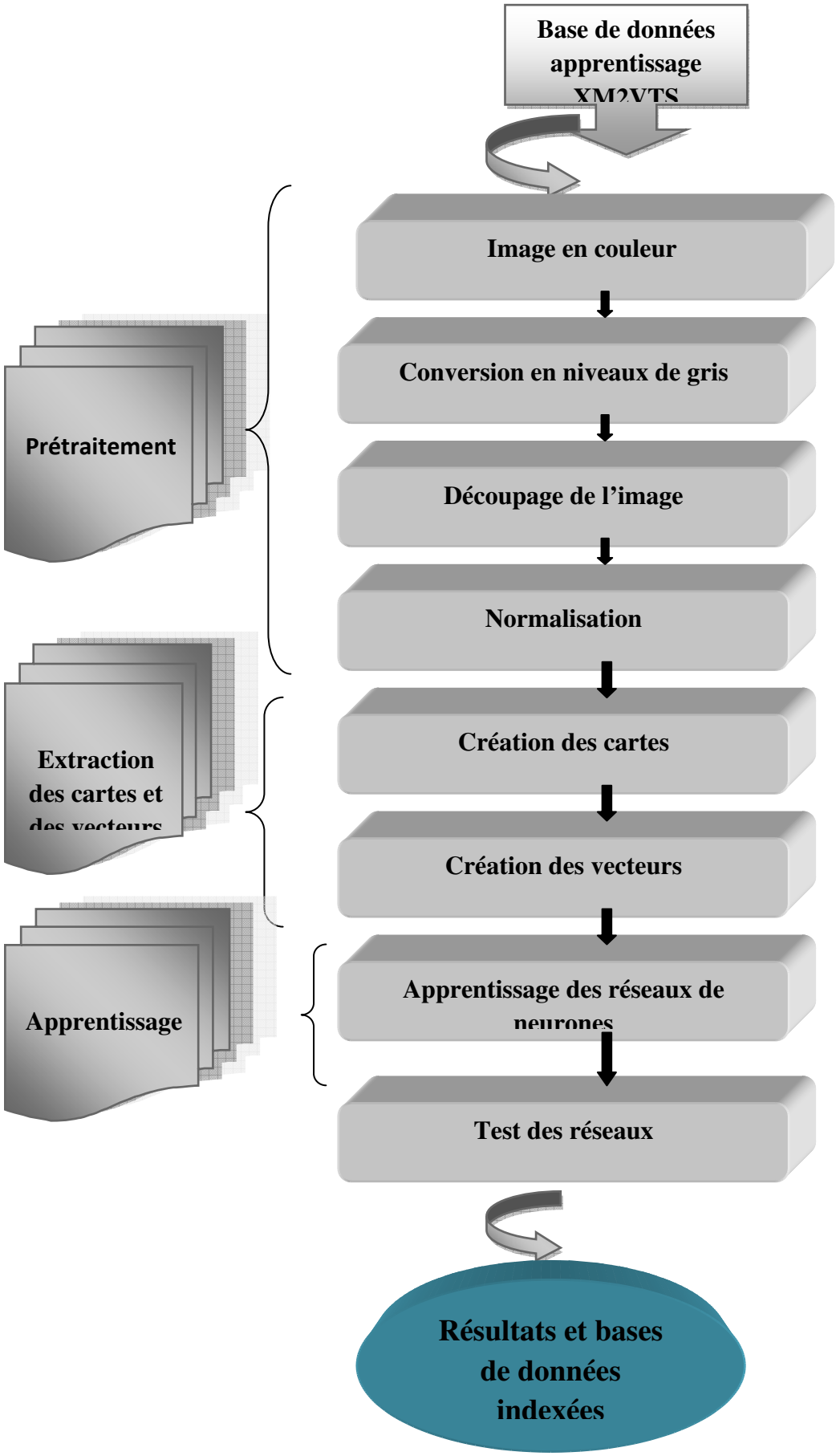
Pour la classe des imposteurs, les 95 imposteurs sont répartis dans deux ensembles : 25 pour l'ensemble d'évaluation et 75 pour l'ensemble de test.

Les tailles des différents ensembles de la base de données selon les deux configurations cités précédemment sont reprises dans le tableau suivant.

**Tableau V.1 :** Répartition des photos dans les différents ensemble

| Ensemble | Clients | Imposteurs |
|---|---|---|
| Apprentissage | 600 (3 par personne) | 0 |
| Evaluation | 600 (3 par personne) | 200 (8par personne) |
| Test | 400 (2 par personne) | 560 (8 par personne) |

## V.3 Architecture fonctionnelle de la détection



### V.3.1. Prétraitement d'image

Le prétraitement est une phase importante dans le processus globale d'identification. C'est une méthode simple qui augmente en général les performances du système. Elle permet souvent une première réduction des données et elle atténue les effets d'une différence de conditions lors des prises de vues.

❖ **Algorithme de prétraitement**

Les opérations de prétraitement sur l'image d'entrée sont collectées dans une seule fonction nommée ***prétraitement ( )*** ; l'algorithme de cette fonction est donnée par :

% Y : image d'entrée (à traiter).

% prétraitement () : fonction de pratiquement.

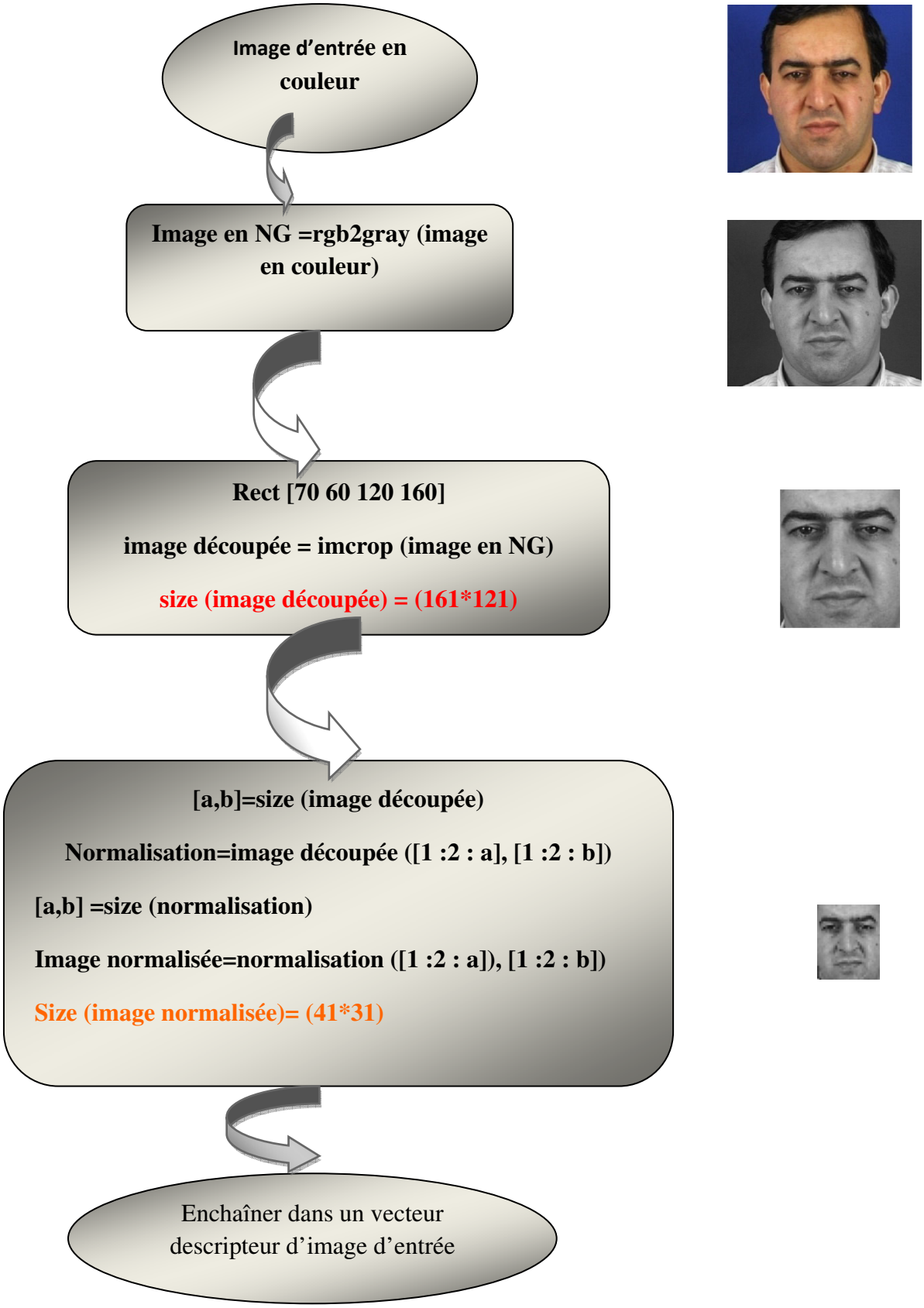
**Prétraitement (Y)**

- Lire Y
- Transférer Y de RGB en niveaux de gris : (Z=rgb2gray(Y))
- Découper Z
- Normaliser Z

- Enchaîner (superposer) les lignes ou les colonnes de Z en (A)
- **Enregistrer A**

**Algorithme 1** L'algorithme de la fonction de prétraitement

## ❖ Organigramme de prétraitement



### a) Transformation des images couleur en NG

La transformation des images couleurs en noir et blanc est nécessaire dans notre projet pour éliminer les paramètres de couleur, la figure ci-dessous représente l'effet de la fonction de"rgb2gray". Cette transformation est nécessaire, car notre travail se base sur des images niveaux de gris. Sinon il faudrait prendre ces paramètres en considération et travailler en couleur donc en trois dimensions ce qui alourdit davantage notre programme.

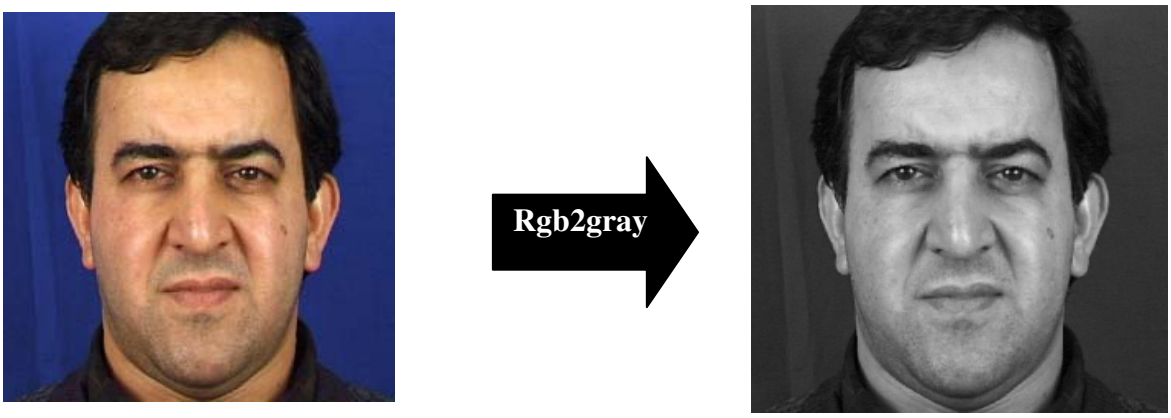



**Figure V.2 :** Le fait du fonction de"rgb2gray"

### b) Découpage des images

En regardant les images nous apercevons directement qu'apparaissent au niveau du cou des particularité non souhaitées comme les cols des chemises etc … des paramètres non informants. Par ailleurs les cheveux sont également une caractéristique changeante au cours du temps. C'est pourquoi nous avons décidé de découper les images verticalement et horizontalement, selon la figure ci-dessous.

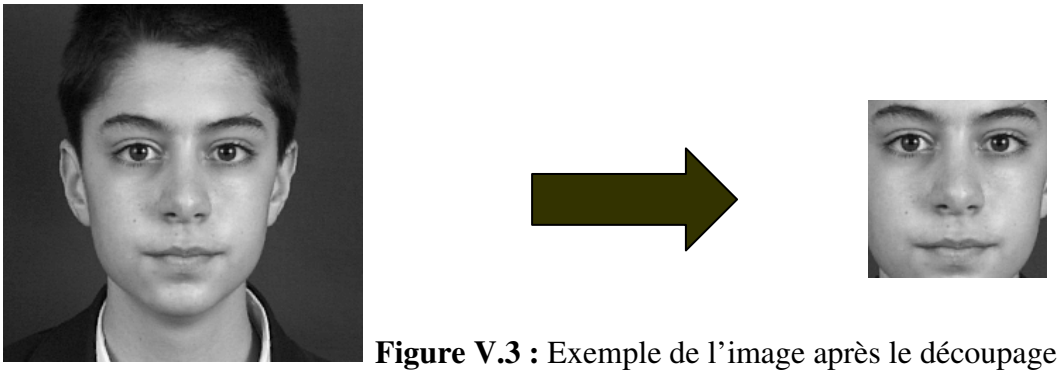


**Figure V.3 :** Exemple de l'image après le découpage

c) **Photo normalisation**

La photo normalisation a un double effet : d'une part elle supprime pour tout vecteur un éventuel décalage par rapport à l'origine et ensuite elle supprime tout effet d'amplification (multiplication par un scalaire). Pour chaque image nous effectuons l'opération suivante :

$$photonormalisation(x) = \frac{x - mean(x)}{std(x)}$$

*std* (*x*) : l'écart type de la variable ***x***.

**V.3.2 Création des cartes et des vecteurs**

% Y : Image d'entrée (à traiter).

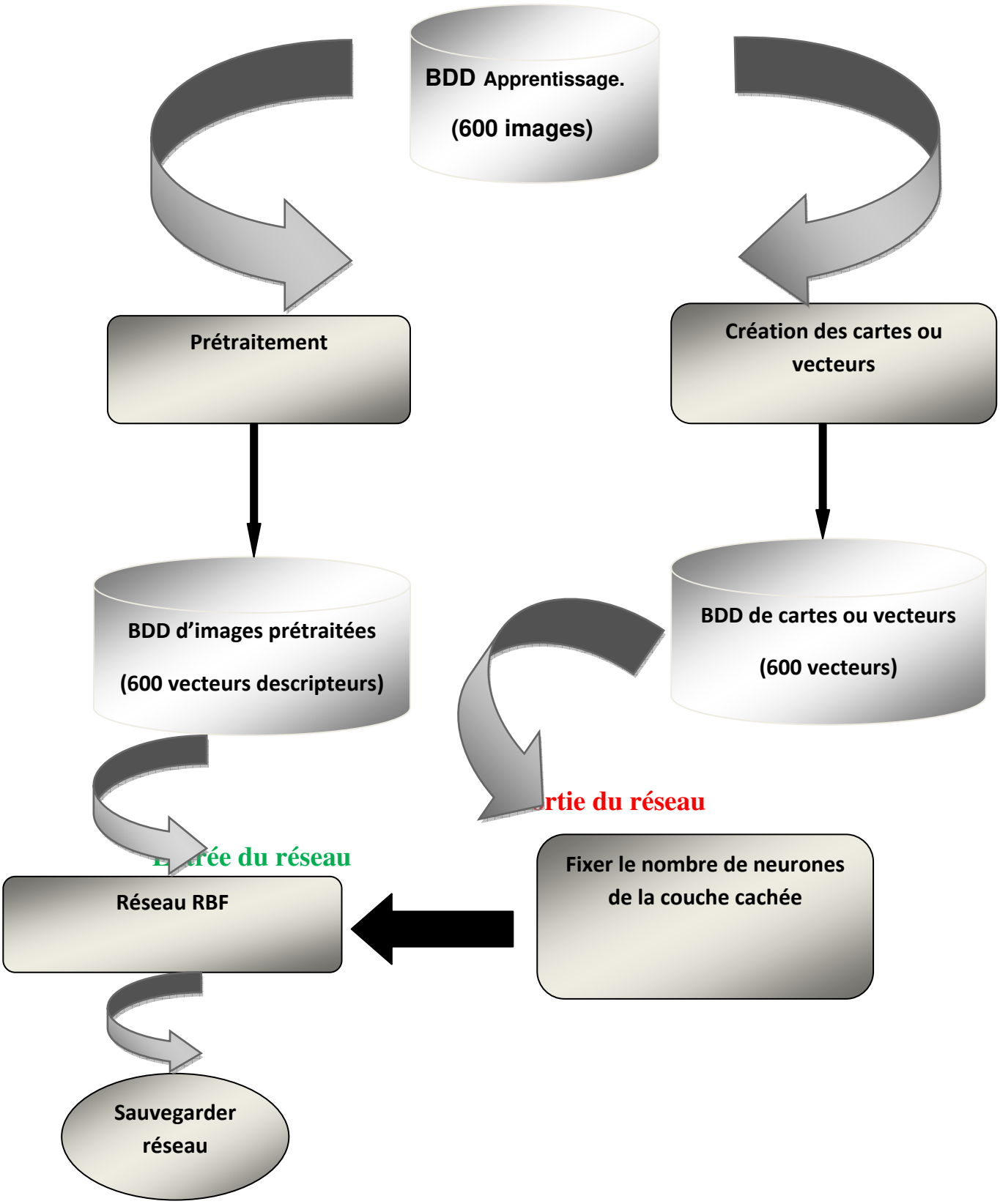
% : Création des cartes ou vecteurs () : fonction de pratiquement.

Création des cartes ou vecteurs de **(Y)**

- Lire Y
- Transférer Y de RGB à niveaux de gris (Z=rgb2gray(Y))
- Découper Z
- Normaliser Z à une image de size (41*31)
- Création d'une matrice F de taille (41*31) ou vecteur V de size(10*1)
- F = +1 : au niveau de caractéristique (centre des yeux, nez,le coin de la bouche)
- F= -1 : ailleurs
- F =F convolé avec un filtre Gaussien (3*3) :(1/16*[1 2 1,2 4 2,1 2 1])
- Où V= les cordonnées caractéristiques (centre des yeux, nez,le coin de la bouche)
- Enchaîner (superposer) les lignes ou les colonnes F dans (A)
- **Enregistrer A ou V**

**Algorihme 2** Algorithme de création des cartes et vecteurs

## V.4 Organigramme d'apprentissage de réseaux RBF



### V.4.1 Apprentissage du réseau (Cas de réalisation de cartes)

Les résultats de l'apprentissage du réseau sont donnés par le **tableau1**et l'erreur quadratique mesurée dans ce cas est présentée par la courbe de la **figure 1.**

| **Nombre de neurones** | **0** | **25** | **50** | **75** |
|---|---|---|---|---|
| **MSE** | **0,00101726** | **0,00101306** | **0,00100913** | **0,001** |

**Tableau V.2 :** Variation de MSE en fonction de nombre de neurone

(**Cartes**)

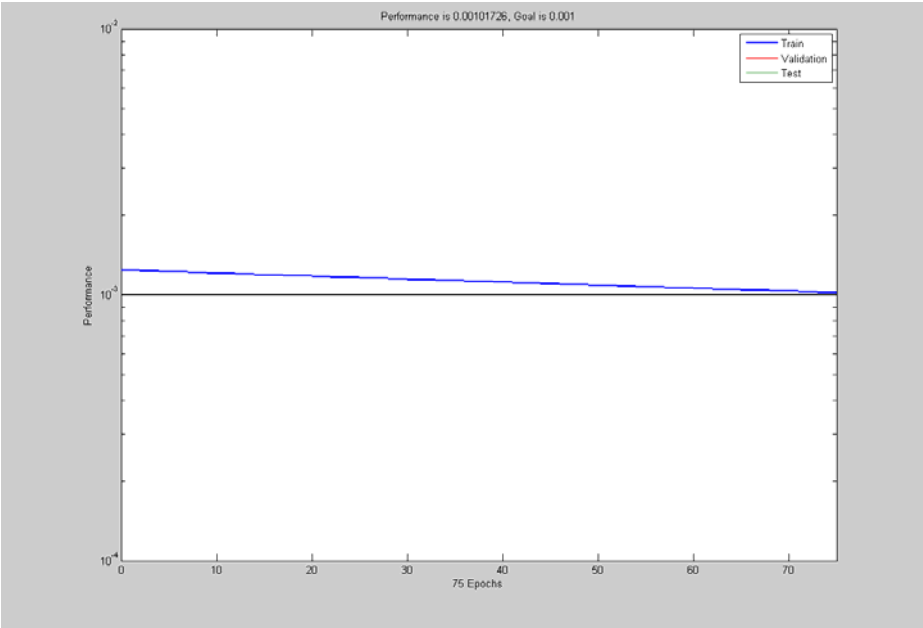



**Figure V.4 :** La variation des erreurs en fonction nombre d'itération dans le cas des cartes

La figure illustre les variations d'erreur quadratique moyenne (**MSE**) en fonction du nombre d'itérations. Elle tend vers la valeur **0,001** et présente une **stabilité** quand le **nombre d'itérations** atteind **75.** Nous observons donc que l'**erreur diminue** avec l'**augmentation de nombre de l'itérations**.

### V.4.2 Apprentissage réseau (Cas de réalisation de vecteurs)

Nous fixons l'erreur à goal l= 1.8 et nous lançons les itérations jusqu'au rapprochement maximum de l'erreur fixé auparavant.

| NR | 0 | 25 | 50 | 75 | 100 | 125 | 150 | 175 | 200 | 225 | 250 | 275 | 300 | 325 | 350 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| MSE | 1,841 | 1,839 | 1 ,836 | 1 ,833 | 1,830 | 1 ,829 | 1,827 | 1,824 | 1,821 | 1,819 | 1,816 | 1,814 | 1,811 | 1,810 | 1 ,8 |

**Tableau V.3 :** Variations de **MSE** en fonction du nombre de neurones

**(Vecteurs)**

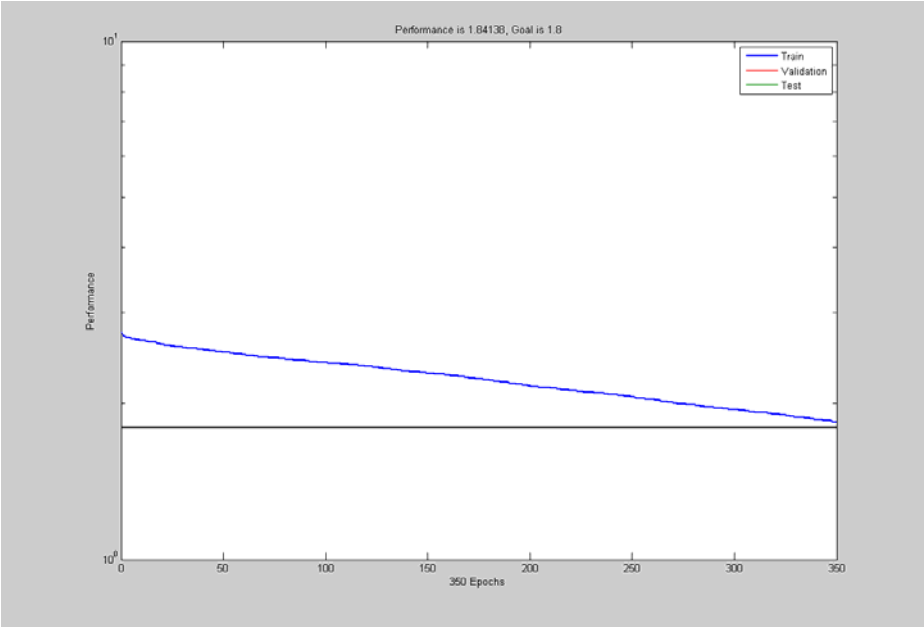



**Figure V.5 :** L'erreur en fonction du nombre d'itérations

(Vecteurs Caractéristiques)

La **figure V.5** illustre les variation de l'erreur quadratique concernant les vecteuts caractéristiques. Cettee dernière est calculée entre les vecteurs réalisés d'une façon manuelle et les vecteurs obtenus par le réseau de neurones.

Son algorithme est donné par l'équation suivante :

**MSE** = E[**e(x)**]

**e(x)** = Y(x)-X(x)

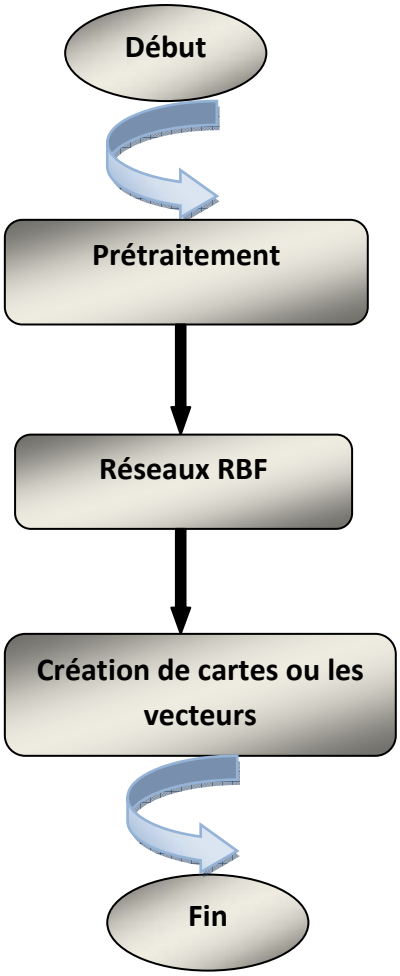
**Y(x)** : la sortie du réseau

**X(x)** : la sortie désirée

### 4.2 Procédure de test

En premier lieu, nous procédons à la lecture de la base de donnée tr**set** par la procédure « lire ». Après, nous effectuons le prétraitement par la procédure de « réduction des images » pour rejeter les paramètre indésirables, puis nous transformons les images en NG par la fonction de « rgb2gray ».Puis nous faisons la normalisation de l'image. Dans la phase finale nous réalisons la création des cartes et des vecteurs (600 cartes et vecteurs) et nous fixons le nombre de couches cachées. Nous avons donc à l'entrée du réseau RBF la BDD originale et nous récupérons à la sortie la BDD des cartes et des vecteurs. Finalement nous sauvegardons le réseau. L'application de notre programme sur la BDD XM2VTS est présenté dans le paragraphe 5 du même chapitre.

Organigramme de test



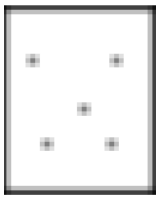
## V.5 Résultats sur images de la BDD XM2VTS

### V.5.1 Application avec création de cartes

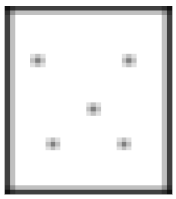
#### V.5.1.1 Cartes manuelles de personnes différentes

Nous avons choisis pour cette étude des images de personnes différentes : Hommes et femmes, port de lunettes, port de moustache, port de barbe… pour mieux tester l'efficacité du choix des points d'intérêts d'une façon manuelle.

**1) Apprentissage** **$Carte_A$**

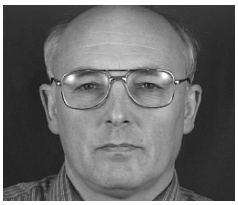

**Test**

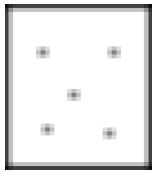



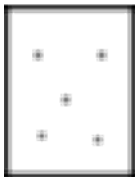
**Face 52** **$Image_n$** **$Image_n$** **$Carte_T$**

Présence de cheveux sur les sourcils n'a pas d'effet sur la détection. Car dans le cas de notre modèle les sourcils ne représente pas une région d'intérêt.

**2) Apprentissage** **Test**





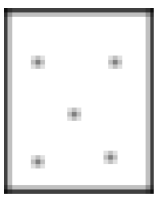


**Face 53** **$Image_n$** **$Carte_A$** **$Image_n$** **$Carte_T$**

Détection parfaite car l'image est bien centrée ,bien découpée et lunettes ne couvrant pas les yeux.

**3) Apprentissage** **Test**



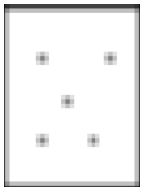


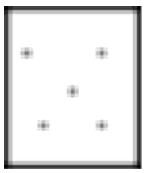
**Face 54** **$Image_n$** **$Carte_A$** **$Image_n$** **$Carte_T$**

Quand l'image est neutre nous remarquons une symétrie des points des yeux et coins de la bouche par rapport au nez. Alors qu'en présence de lunettes ou moustaches cette symétrie n'existe plus. Exemple : les cas (port de lunettes ; port de barbe ; port moustaches )

## 4) Apprentissage Test

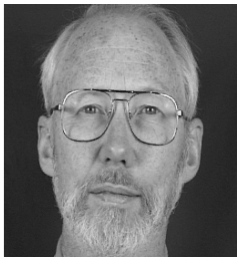


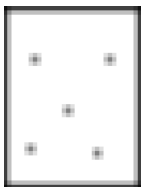


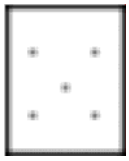


**Face 58** **Image$_n$** **Carte$_A$** **Image$_n$** **Carte$_T$**

## 5) Apprentissage Test





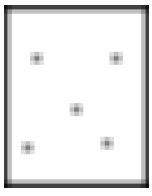


**Face 61** **Image$_n$** **Carte$_A$** **Image$_n$** **Carte$_T$**

L'inclinaison du visage entraine un découpage non parfait ( la photo n'est pas centrée). Et ceci perturbe la détection. Même remarque pour la photo 58. La tête n'est pas inclinée mais l'image n'est pas bien centrée ce qui entraine un découpage plus accentué sur le côté gauche, ce qui perturbe la détection.

## 6) Apprentissage Test



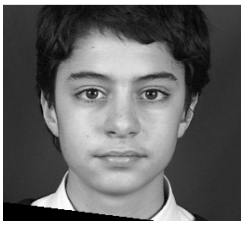


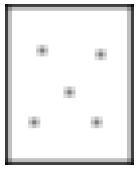
**Face 65** **Image$_n$** **Carte$_A$** **Image$_n$** **Carte$_T$**

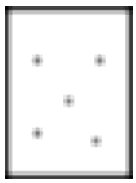
Les lunettes couvrant les yeux et moustache couvrant bouche côté gauche. Le port de lunettes influe sur la détection toutes les images présentées ci-desssus le montrent bien.

## 7) Apprentissage

## Test





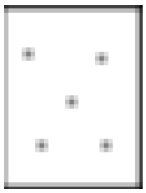


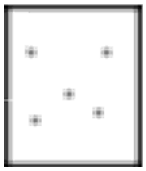
**Face 68** **Image$_n$** **Carte$_A$** **Image$_n$** **Carte$_T$**

## 8) Apprentissage

## Test







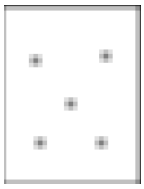
**Face 72** **Image$_n$** **Carte$_A$** **Image$_n$** **Carte$_T$**

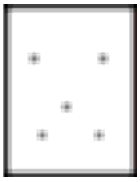
La présence de cheveux sur les sourcils n'influe pas sur la détection comme nous le montre bien l'image 72 surtout les pts d'intérêts du côté gauche car le regard est dirigé vers la gauche.

## 9) Apprentissage

## Test



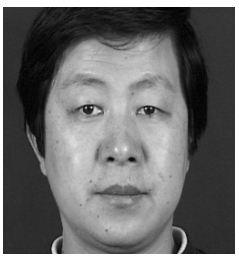


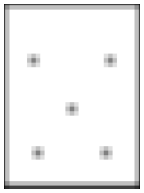


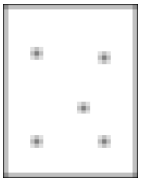
**Face 116** **Image$_n$** **Carte$_A$** **Image$_n$** **Carte$_T$**

## 10) Apprentissage

## Test



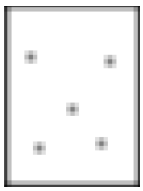




**Face 108** **Image$_n$** **Carte$_A$** **Image$_n$** **Carte$_T$**

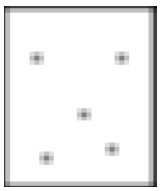
## 11) Apprentissage





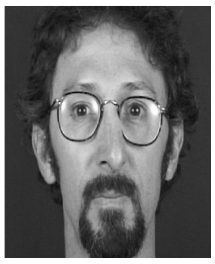
**Face 135** **Image$_n$** **Carte$_A$**

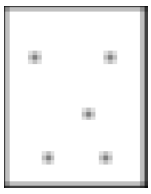
## Test

**Image$_n$** **Carte$_T$**

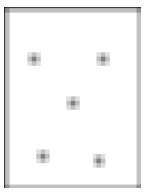
## 12) Apprentissage





**Face 191** **Image$_n$** **Carte$_A$**

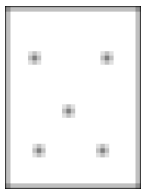
## Test

**Image$_n$** **Carte$_T$**

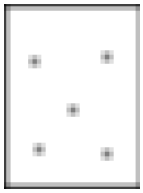
## 13) Apprentissage





**Face 209** **Image$_n$** **Carte$_A$**

## Test

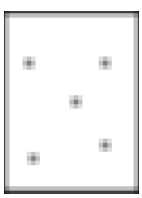



**Image$_n$** **Carte$_T$**

Détection parfaite pour les images (68 ;116 ;108 ;209) il s'agit d'images neutres sans port de lunettes. Malgré la présence de moustache et la barbe sur l'image 209 nous constatons que ça n'influe aucunement pas. Car comme nous l'avons mentionné la barbe n'a pas ou peu d'effet alors que dans ce cas particulier la moustache est dégagée (la bouche n'est pas couverte). Donc, les points d'intérêts de la bouche est correctement détectée.

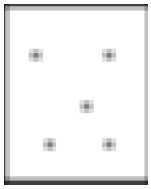
## 13) Apprentissage





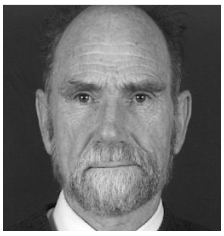
**Face 249** **Image$_n$** **Carte$_A$**

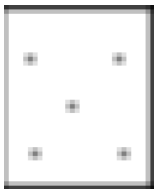
## Test

**Image$_n$** **Carte$_T$**

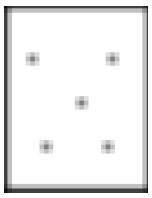
## 14) Apprentissage





**Face 259** **Image$_n$** **Carte$_A$**

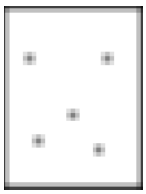
## Test

**Image$_n$** **Carte$_T$**

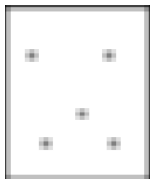
## 15) Apprentissage





**Face 264** **Image$_n$** **Carte$_A$**

## Test

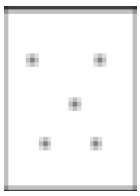



**Image$_n$** **Carte$_T$**

Sur ces trois images nous pouvons dire que la barbe influe peu( image 259). Alors que la moustache est détectée car à notre avis elle se superpose à la bouche. Donc elle peut erroner la détection des coins de la bouche.(image 264). Le mouvement du visage peut aussi avoir de l'effet sur la détection (image 249).

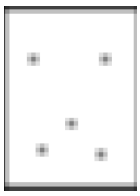
## 16) Apprentissage





**Face 267** **Image$_n$** **Carte$_A$**

## Test

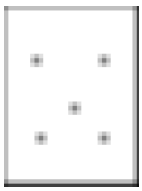



**Image$_n$** **Carte$_T$**

Moustache dégagée et grandes montures.

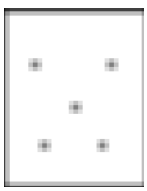
## 17) Apprentissage





**Face 357** **Image$_n$** **Carte$_A$**

## Test

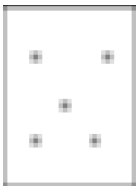
**Image$_n$** **Carte$_T$**

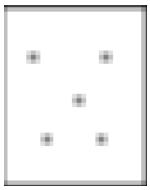
Lunettes non détecté (357) car monture ne couvre pas les points d'intérêts (Monture très grandes).

## 18) Apprentissage

## Test





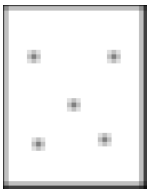


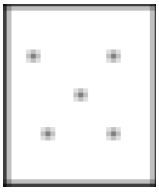
**Face 336** **Image$_n$** **Carte$_A$** **Image$_n$** **Carte$_T$**

## 19) Apprentissage

## Test







**Face 92** **Image$_n$** **Carte$_A$** **Image$_n$** **Carte$_T$**

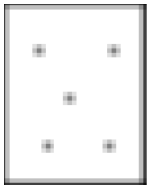
Nous constatons d'après les cartes obtenues que les différents aspects dans les images sont détectés.

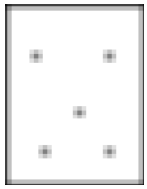
### V.5..1.2 Cartes manuelles de la même personne

## 1.1) Apprentissage

## Test





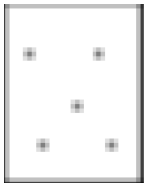


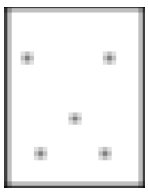
**Face 333.1** **Image$_n$** **Carte$_A$** **Image$_n$** **Carte$_T$**

## 1.2) Apprentissage

## Test





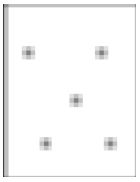


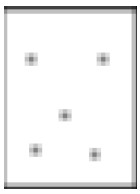
**Face 333.2** **Image$_n$** **Carte$_A$** **Image$_n$** **Carte$_T$**

## 1.3) Apprentissage

## Test







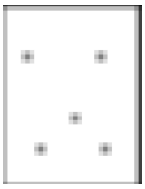
**Face 333.3** **Image$_n$** **Carte$_A$** **Image$_n$** **Carte$_T$**

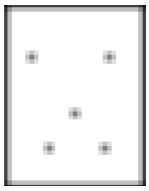
Le mouvement de la tête vers le haut comme dans les images 333.1 et 333.2 montrent que ça influe surtout sur la détection du nez. Alors que dans les images 333.2 et 333.3 la déviation du regard vers la gauche est repéré par un léger décalage sur les cartes respectives. Cela est plus significatifs sur l'image 333.1 où les yeux st levés vers le haut.

## 2.1) Apprentissage

## Test





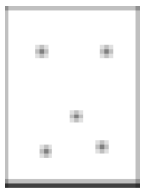


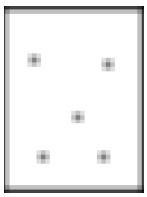
**Face 222.1** **Image$_n$** **Carte$_A$** **Image$_n$** **Carte$_T$**

## 2.2) Apprentissage

## Test





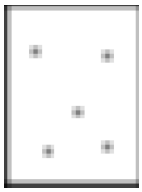


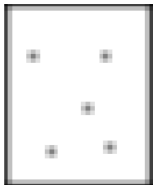
**Face 222.2** **Image$_n$** **Carte$_A$** **Image$_n$** **Carte$_T$**

## 2.3) Apprentissage

## Test







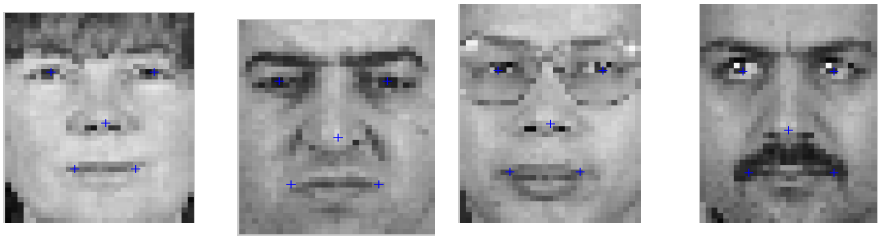
**Face 222.3** **Image$_n$** **Carte$_A$** **Image$_n$** **Carte$_T$**

Dans les images 222.1 et 222.2 notre détecteur repère bien la différence d'expression de visages. Le mouvement des yeux est repéré sur les images 222.2 et 222.3.

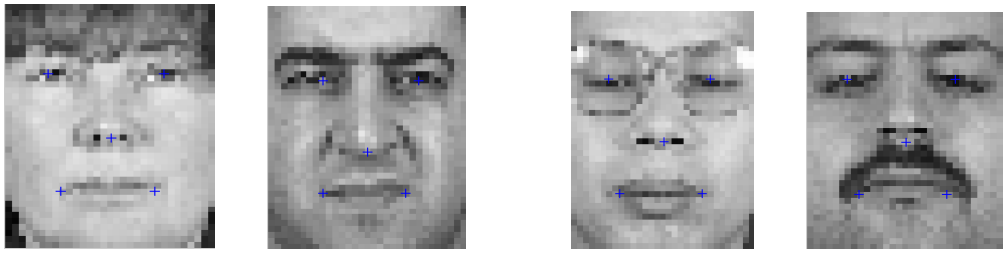
### V.5.2 Application avec création de vecteurs

#### V.5.2.1 Création de vecteurs de personnes différentes

## 1) Apprentissage

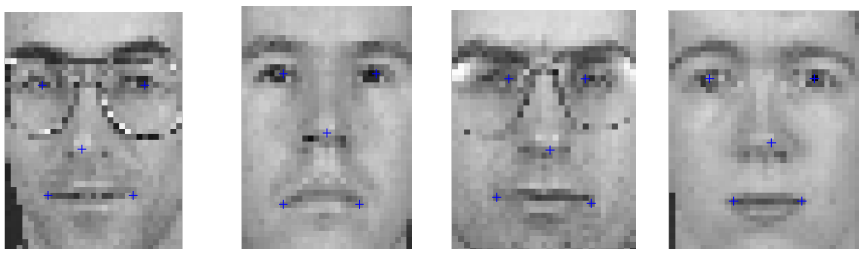


## Test

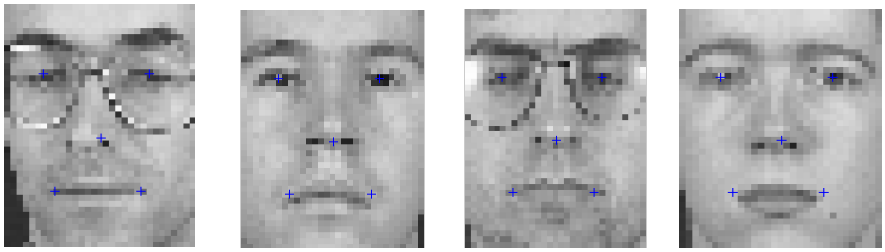


Toute la série de photo est bien centrée. A part la première photo qui est neutre les trois autres présentent des variations : Expression dans la deuxième, port de lunettes et regard vers le bas dans la troisième et port de moustache et regard vers le nez dans la quatrième. Malgré cela l'apprentissage est bien fait. On peut dire que la détection reste bien réussie dans le test. La première image est la mieux détectée dans l'apprentissage. Alors que la phase test est meilleure dans le cas des trois autres images. Ceci peut être dû au fait que la pemière image est bien centrée et neutre. Et le reste des photos présentent des variabilités. Dans l'ensemble la détection est bonne dans l'apprentissage. Elle est satisfaisante dans le test.

## 2) Apprentissage



## Test

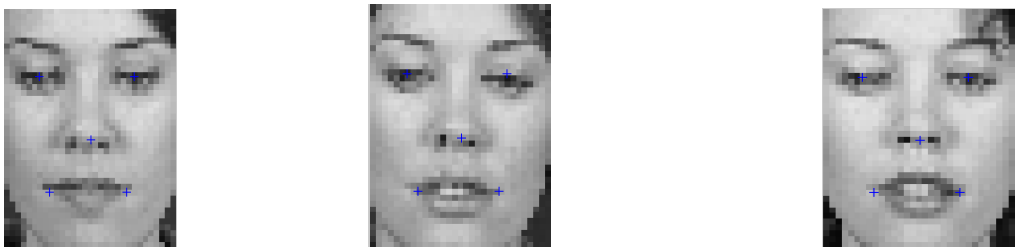


Nous remarquons que l'apprentissage est bien fait comme dans tous les cas en général. Dans la phase de test le nez est bien détecté quand la personne est face à l'appareil d'acquisition. S'il y'a une légère orientation de la tête le nez est mal détectée. On peut direque la bouche est correctement détectéeencore une fois quand la personne est face à l'appareil. Dans le cas de l'orientation (1ère photo) ou l'expression (4ème photo) la bouche a des difficultés à être détectéee.

### V.5.2.2 Création de vecteurs de la même personne

## Apprentissage (vecteurs )

## Test (vecteurs)



La 1ère photo et la 3ème photo sont bien détectées dans le test. La 2ème est mieux détectée en apprentissage.

## Discussion

Les yeux sont bien détectés dans tous les cas et dans les deux BDD. A partir des résultats apprentissage et test nous pouvons dire que la détection varie d'un ensemble à un autre. Nous pouvons dire que l'apprentissage est bon dans tous les cas de variation dans le visage et même le mouvement de la tête. Alors que dans le cas du test le problème ne se pose pas pour les faces centrées et neutres. Mais dans l'ensemble la détection reste assez bonne dans le test. Mais la détection n'est pas efficace sur tous les points du visage par exemple : l'apprentissage est mieux pour la détection des yeux et la bouche, mais le nez le test est mieux sollicité. Dans le cas d'une même nous cherchons à tester l'efficacité du détecteur face au mouvement des régions d'intérêt sur le visage d'une même personne. L'apprentissage est parfait et dans le test les yeux et le nez ne posent aucun problème, néanmoins quelques légères difficultés à détecter la bouche. Nous pouvons finalement dire que les yeux sont bien détectés dans les deux BDD et ceci en présence ou absence de variations. Le nez est sensible au mouvement de la tête, donc il est le mieux détecté dans les images bien centrée, face à l'appareil et neutre. La difficulté reste celle de la détection de la bouche et ceci est sûrement dû au nombre de points utilisés pour le caractériser.

## Conclusion

D'après notre étude et toutes les expériences réalisées nous pouvons dire qu'il ya des régions très efficaces dans le visage. Qui nous permettent de diminuer les points détectés dans le visage et ces points donnent de meilleurs résultats. Comme nous le montrent nos expériences sur les yeux, le nez et les coins de la bouche et nous obtenons de bons résultats avec une erreur quadratique **MSE=0,001** pour les carte et **MSE=1,8** pour les vecteurs. Nous pouvons affirmer que le réseau pour la création de cartes présente une meilleure erreur. Mais le réseau pour création de vecteurs fournit une meilleure détection et une représentation vectorielle facile à traiter.

# Conclusion générale

La détection et la reconnaissance d'individus demeurent des problèmes complexes, malgré les recherches actives actuelles. Il y a de nombreuses conditions réelles, difficiles à modéliser et prévoir, qui limitent les meilleurs systèmes actuels.

Le système de détection proposé présente une alternative simple mais robuste. La détection est réalisée par une méthode hybride alliant traitement de l'image et réseau de neurone.

L'approche la plus répandue est sans doute, celle utilisant **un réseau de neurones** pour classifier les pixels de l'image, en tant que visage ou non-visage.

L'inconvénient de cette approche réside dans le temps de calcul qui ne permet pas souvent de faire des traitements en temps réel.

Nous pouvons dire que les réseaux de neurones sont des modèles de calculs apprenant, généralisant et organisant des données.

Cette application des réseaux de neurones montre la simplicité avec laquelle, exemples en main, cet outil est utilisable pour obtenir rapidement un premier résultat. Cependant, ce sont la précision de la détection et le taux d'erreurs qui sont sujets à la plupart des travaux. L'efficacité et la robustesse d'un système se jugeront surtout sur les heuristiques et les méthodes employées pour l'optimiser. Ces méthodes doivent d'une part minimiser le taux de mauvaises détections, et d'autre part maximiser le score de bonnes détections. Ces deux scores ne sont pas totalement complémentaires, nous avons en effet discuté des dangers de l'heuristique éliminant toute détection en chevauchant une autre plus probable. Ainsi, l'erreur se définit à la fois en fonction du nombre de mauvaises détections, mais aussi du nombre de détections omises. C'est ce balancement entre ces deux facteurs qui rend difficile l'optimisation. Les méthodes sont en général efficaces pour l'un mais présentent des effets de bords pour l'autre. C'est ainsi que traiter l'ensemble des cas peut complexifier exponentiellement le traitement pour une amélioration parfois logarithmique.

De plus, si l'on considère tous les angles, en trois dimensions, sous lesquels peuvent être présenté un visage, la reconnaissance devient beaucoup moins triviale. L'utilisation de la symétrie et de la forme du visage peut amener à déduire depuis un visage vu de côté sa forme frontale. profil droit. Cependant, ces angles, bien que les plus courants, ne sont pas les seuls.

Nous avons présenté aussi la localisation de régions d'intérêts de visage et l'authentification d'individus, ainsi que les différentes techniques utilisées. Si la biométrie est un enjeu important au niveau économique, la recherche, en particulier dans le domaine de la reconnaissance des personnes à partir d'une image 2D de leur visage, offre encore un champ d'investigations très ouvert

Nous pouvons dire que les réseaux de neurones sont des modèles de calculs apprenant, généralisant et organisant des données.

Les yeux sont bien détectés dans tous les cas et dans les BDD. A partir des résultats apprentissage et test nous pouvons dire que la détection varie d'un ensemble à un autre. Nous pouvons dire que l'apprentissage est bon dans tous les cas de variation dans le visage et même le mouvement de la tête. Alors que dans le cas du test le problème ne se pose pas pour les faces centrées et neutres. Mais dans l'ensemble la détection reste assez bonne dans le test. Mais la détection n'est pas efficace sur tous les points du visage par exemple : l'apprentissage est mieux pour la détection des yeux et la bouche, mais le nez le test est mieux sollicité. Dans le cas d'une même nous cherchons à tester l'efficacité du détecteur face au mouvement des régions d'intérêt sur le visage d'une même personne. L'apprentissage est parfait et dans le test les yeux et le nez ne posent aucun problème, néanmoins quelques légères difficultés à détecter la bouche. Nous pouvons finalement dire que les yeux sont bien détectés dans les deux BDD et ceci en présence ou absence de variations. Le nez est sensible au mouvement de la tête, donc il est le mieux détecté dans les images bien centrée, face à l'appareil et neutre. La difficulté reste celle de la détection de la bouche et ceci est sûrement dû au nombre de points utilisés pour le caractériser

D'après notre étude et toutes les expériences réalisées nous pouvons dire qu'il ya des régions très efficaces dans le visage. Qui nous permettent de diminuer les points détectés dans le visage et ces points donnent de meilleurs résultats. Comme nous le montrent nos expériences sur les yeux, le nez et les coins de la bouche et nous

obtenons de bons résultats avec une erreur quadratique **MSE=0,001** pour les carte et **MSE=1,8** pour les vecteurs. Nous pouvons affirmer que le réseau pour la création de cartes présente une meilleure erreur.  Mais le réseau pour création de vecteurs fournit une meilleure détection et une représentation vectorielle  facile à traiter.

## Perspectives

- On devrait utiliser plus de points pour caractériser le nez et la bouche, pour mieux les détecter en présence de variation.
- L'information de couleur est un outil efficace pour identifier des zones du visage. Cependant, cela n'est pas utilisable lorsque le spectre de couleurs varie de manière significative entre l'arrière plan et le visage
- La localisation et le suivi de visages dans des séquences d'images sont sollicitées .

# Liste des figures

# Liste des tableaux

# Sommaire

# Bibliographie


[1] Diane FERGUSON ,Français LEVASSEUR ,Thomas Sensible, Wilfried TALPIN. Détection de visages, LAPI .ISI CO 2004.

[2] MECHALSI Mohamed Réda, OUAFEK Redouane .La Détection des Visages dans une image ,Université Med Khi der Biskra soutenue en 2008/2009.

[3] E. Helms et B. K. Low. "*Face detection : A survey*", Computer Vision and Image Understanding, vol. 83, no. 3, pp. 236-274, 2001.

[4] F. Zernike. "*Diffraction theory of the cut procedure and its improved form, the phase contrast method"*. Physic, 1:pp. 689-704, 1934.

[5] M. Turk, A. Pent land, *Eigen faces for recognition*. J. of Cognitive Neuroscience 3, 72–86, 1991.

[6]. J. Haddadnia, M. Ahmadi, and K. Faze, "*An Efficient Feature Extraction Method with Pseudo Zernike Moment in RBF Neural Network Based Human Face Recognition System*", EURASIP JASP, vol. 9, pp. 890-891, 2003.

[7] P. Belhumeur, J. Hispania, D. Krieg man, *Eigen faces vs. Fisher faces: Recognition Using Class Specific Linear Projection*, IEEE Transactions on Pattern Analysis and Machine Intelligence 19, pp. 711-720, 1997.

[8] D. L. Sweets, J. Wing, *Using discriminate eigenfeatures for image retrieval*, IEEE Trans. Patt. Anal. Mach. Intel. 18, 831– 836, 1996.

[9] Kha chi Djamel ,Ben bouta Zakaria. Introduction à La calcul La biolite éditions , WOL per p,1991 ,ISBN2-7296-0372-7.

[10] Diagnostic des défauts d'un système Industriel par les Réseaux de Neurones,2007

[11] Clarke A. C. 2001,L'odyssée de L'espace. Volume 349.j'AILU,1968.

[12] D'aval E,P, Naim, " Des réseaux de neurones" Deuxièmes Edition,Eyrolles,1993.

[13] Jastrun R."Aude là du cerveau" Pluriel,1982.

[14] Lallemand Y. Intégration neurone-Symbolique et intelligence artificielle: Application et implémentation parallèle . PhD thése,Université Henri Poincaré ,Nancy I, Juin 1996.

[15] McCulloch W .C. PittsW. "A logical calculus of the ideas immanent in nervous activity", ",page 115-133. Bulletin of mathematical biophysics,vol.5,1943.
[16] I. Rivals,L . Personna Z G . Dreyfus , " modélisation, classification et commande par réseaux de neurones: principes fondamentaux, méthodologie de conception et illustrations industrielles", Ecole supérieure de physique et de chimie industrielles de la ville de paris laboratoire d'électronique 10, rue Vauquelin 75005 PARIS.
[17] Pélissier A and A . Tète .Sciences Cognitives: textes fondateurs ( 1943- 1950) psychologie et sciences de la pensée. PUF , 1995 ISBN 2-13-46695-8.
[18] Détection de Visages à L'Aide de Réseaux de Neurones, 2005.
[19]SENECHAL Thibaud . Détection et extraction des éléments caractéristiques du visage, 2007-2008.
[20] S. Sahli, R. Boudraa. Reconnaissance de visages par les ondelettes optimales. Université Mohamed khider biskra. Département électronique. Juin 2009.
[21] Y. Tahraoui, A. Tounsi. Analyse de visage par ondelettes associées aux réseaux de neurones. Université Mohamed khider biskra. Département électronique. 2007.
[22] S . Bedra, N . Mansoura. identification et authentification des visages par FLDA et ACP. Université Mohamed khider Biskra. Département électronique. 2007.
[23] A. Ouamane, A. Mehdaoui. identification et authentification des visages en biométrie (transformation de houg et FLDA). Université Mohamed khider Biskra. Département électronique. 2009.
[24] Analyse de textures par ACP. Université de Batna. Département électronique. 1994.