UNIVERSITÉ DU QUÉBEC

MÉMOIRE PRÉSENTÉ À
L'UNIVERSITÉ DU QUÉBEC À TROIS-RIVIÈRES

COMME EXIGENCE PARTIELLE
DE LA MAÎTRISE EN MATHÉMATIQUES RT INFORMATIQUE APPLIQUÉES

PAR
LAMRI LAOUAMER

APPROCHE EXPLORATOIRE SUR LA CLASSIFICATION APPLIQUÉE AUX IMAGES

AVRIL 2006

Université du Québec à Trois-Rivières

Service de la bibliothèque

Avertissement

L'auteur de ce mémoire ou de cette thèse a autorisé l'Université du Québec à Trois-Rivières à diffuser, à des fins non lucratives, une copie de son mémoire ou de sa thèse.

Cette diffusion n'entraîne pas une renonciation de la part de l'auteur à ses droits de propriété intellectuelle, incluant le droit d'auteur, sur ce mémoire ou cette thèse. Notamment, la reproduction ou la publication de la totalité ou d'une partie importante de ce mémoire ou de cette thèse requiert son autorisation.

REMERCIEMENT

Je voudrais remercier fortement toutes les personnes ayant contribué à la réalisation de ce mémoire, et m'ayant soutenu durant toutes mes années d'études.

Je remercie particulièrement :

- Mon directeur de recherche, le professeur Ismaïl Biskri pour son encadrement, sa patience, ses conseils et ses encouragements constants.

- Aux professeurs Mhamed Mesfioui et François Meunier pour avoir accepté l'évaluation de mon mémoire .

- Tous les professeurs et étudiants à la maîtrise du département de mathématiques et informatique appliquées pour leur amitié sincère.

- À notre secrétaire Chantal Lessard, pour sa générosité et sa disponibilité permanente.

DÉDICACES

Le fruit des mes études en maîtrise est dédié à :

- Toute ma famille, particulièrement ma mère. Sans son amour et son support au fils des années, ce mémoire n'existerait pas.

- À ma femme Radhia pour son soutien permanent.

- Mes frères et sœurs : Hamoudi, Rachid, Salim, Salima, Zahia, Yasmina, Samira, Sabrina, Samia et Djezia.

- À tous les petits de la famille : Rayan, Linda, Lina, Salah El Dine, Imen, Meriem, Chahla et Mouna.

- À tous mes amis(es) sans exception.

RÉSUMÉ

Dans ce mémoire, nous avons proposé une méthode semi-automatique de classification pour les images numériques, l'approche ayant déjà fait ses preuves dans la classification textuelle et le traitement de la parole. La méthode proposée est celle des N-Grams. La principale caractéristiques de cette approche est qu'elle est indépendante de la nature de l'information à traiter, voire indépendante de l'encodage même de l'information. Dans le cas de la classification appliquée aux images, la notion des N-Grams est basée sur les valeurs d'intensités des pixels et plus particulièrement les combinaisons adjacentes de ces valeurs d'intensités et leurs fréquences d'apparitions. Ces deux informations forment le noyau d'informations pour notre classifieur à titre de données d'entrées. Notre classifieur est un réseau de neurone de type ART.

À cet effet, SYCLIM (Système de Classification des Images) est le prototype que nous avons développé pour classifier nos images. Les images, sont toutes au format JPEG avec une résolution de 320*200. Certaines sont en couleur, d'autres aux niveaux de gris. Ces images ont subi plusieurs pré-traitements, tel que la conversion des images couleurs aux niveaux de gris et le filtrage (lissage) pour la diminution du bruit. L'évaluation a porté sur deux bases de données, la deuxième est une extension de la première.

TABLE DES MATIÈRES

CHAPITRE IV : APPROCHE PAR LES N-GRAMS : ÉTAT DE L'ART

CHAPITRE V : RÉSULTATS ET INTERPRÉTATIONS

LISTE DES FIGURES

LISTE DES TABLES

CHAPITRE
I

Introduction

L'information de nos jours prend plusieurs formes. L'information textuelle est la plus répandue certes. Cependant avec l'essor de l'Internet et des outils multimédias, l'information textuelle n'est plus la seule à véhiculer la connaissance. Le son et surtout l'image prennent de plus en plus d'importance. Ne dit-on pas que dans certains cas une image vaut mille textes ? Cet état de fait a pour conséquence une nécessité de plus en plus perceptible de développer des outils à même de permettre de traiter l'image, de l'indexer, de la retrouver dans une base de données, de reconnaître sa forme, etc.

Un rapide tour de l'état de l'art en vision artificielle nous apprend par exemple que l'indexation des images fait actuellement l'objet de recherches très abondantes dans le domaine du traitement d'images et de la vision par ordinateur. Il y est proposé plusieurs méthodes pour associer à une image un ensemble de descripteurs de son contenu, dans le but de mesurer la ressemblance avec les descripteurs correspondant à une requête. On peut citer à titre d'exemple les approches par requête [JOS 98], par exemples et contre exemples [JOS 98], par navigation [JOS 01] etc. Certaines méthodes combinent plusieurs approches comme l'approche par l'exemple avec celle par navigation.

L'indexation d'images a pour but de pouvoir retrouver une image. Une première technique consiste à annoter les images, c'est à dire à leur associer un petit texte ou plusieurs mots clés sur lesquels on effectuera les recherches par la suite. Une autre voie consiste à rechercher directement les images à partir de leur contenu même et non plus de données ajoutées. Cet objectif se décline en deux familles de problèmes de complexités différentes :

- être capable suite à une indexation de retrouver une image à partir d'une image pratiquement identique ou simplement à partir d'un seul fragment.

• être capable de retrouver des classes similaires, même si les images sont très différentes sur le plan du signal: paysage de montagne, etc.

Sur cette base, une méthode de classification est primordiale pour créer des classes de similarité. À cet effet, plusieurs méthodes de classification ont été proposées à ce jour elles sont inspirées à partir de techniques très connues en reconnaissance de formes et en vision par ordinateur.

En général les méthodes de classification s'exécutent en plusieurs étapes. L'étape la plus importante consiste à élaborer des règles de classification à partir de connaissances disponibles à priori ; il s'agit de la phase d'apprentissage. Cette dernière utilise un apprentissage soit déductif soit inductif. Les algorithmes d'apprentissage inductif dégagent un ensemble de règles (ou de normes) de classification à partir d'un ensemble d'exemples déjà classés. Le but de ces algorithmes est de produire des règles de classification afin de prédire la classe d'affectation d'un nouveau cas. Parmi les méthodes de classification utilisant ce type d'apprentissage, on cite les méthodes des k plus proches voisins, la méthode bayésienne, la méthode d'analyse discriminante, l'approche des réseaux de neurones et la méthode d'arbre de décision. Dans les algorithmes d'apprentissage déductif, les règles d'affectation sont déterminées à priori par l'interaction avec le décideur, ou l'expert. À partir de ces règles on détermine les classes d'affectation des objets. Parmi les méthodes utilisant ce type d'apprentissage, signalons à titre d'exemple les systèmes experts et les ensembles approximatifs. De même, certains problèmes de classification nécessitent de combiner les deux types d'apprentissages (inductif et déductif). C'est le cas par exemple des problèmes de défaillances des machines ou du problème de diagnostic dans les images médicales.

D'autre part, la complexité des algorithmes utilisés reste un facteur qui pose d'énormes problèmes. Il n'est pas évident de déterminer dans des temps raisonnables par exemple l'unité d'information qui va nous permettre de parcourir le contenu d'une image. Et plus on dispose de temps pour un traitement plus celui-ci a des chances de déboucher sur des résultats pertinents. Dans le cas contraire les résultats n'auraient aucun intérêt. Ainsi, par exemple, dans la détermination du contour d'une image par une approche polygonale, moins de temps consacré à cette tâche il y a, moins le contour s'identifiera à l'image. Et donc tout traitement fondé sur cette approche serait fatalement inintéressant.

rapport à nos précédents travaux. Une description de notre prototype et l'évaluation de résultats obtenus sera également présentée dans le chapitre 05 de ce mémoire.

Notre mémoire est organisé en cinq chapitres. Dans le premier, nous présenterons la problématique de la classification des images en présentant un résumé sur les techniques de classification les plus utilisées, son application à l'indexation d'images, les limites des ces méthodes et les contraintes. Ensuite, nous exposons l'approche des N-Grams et ses preuves dans le traitement de la l'analyse textuelle. Cette notion est notre approche dans ce mémoire pour le but de classifier nos images.

Dans le second chapitre, nous nous intéressions à la méthode de classification qui consiste à affecter les objets d'un ensemble de données à des catégories ou classes prédéfinies. Un panorama des méthodes de classification est présenté, nous expliquerons les méthodes de classifications les plus connues et qui se résument dans deux catégories : Les méthodes de classification automatique (appelées aussi méthodes de clustering) qui sont des méthodes basées sur la notion d'apprentissage non supervisé(méthodes hiérarchiques et non hiérarchiques), Les méthodes d'affectation (aussi appelées «classificateurs») basées sur la notion d'apprentissage supervisé : méthodes utilisant un ensemble d'exemples où les classes d'appartenance sont connues au préalable. Une argumentation sur les avantages et les inconvénients de chaque méthode est aussi présentée dans ce chapitre.

Le chapitre trois a pour but d'expliquer ce qui se fait par rapport à la vision artificielle et au traitement d'images. Nous présenterons les principaux thèmes de ce domaine en mentionnant l'intérêt de l'analyse et du traitement d'une image, la conversion d'une image en données objets ou plus explicitement identifier les objets contenus dans l'image par l'extraction et l'analyse de caractéristiques abstraites (*features*) à partir des pixels suivant un processus de reconnaissance de forme similaire à celui opéré par l'humain. Nous définissons les différents types d'images, les pré-traitements que subissent une image, la segmentation des images et leurs vectorisation. La notion de vectorisation d'image consiste à représenter une image sous forme vectorielle, autrement dit donner une représentation en vecteurs pour les différentes formes géométriques

constituant l'image. Dans notre approche nous donnons une nouvelle forme de vectorisation d'images à travers l'approche des N-Grams.

Le chapitre quatre portera essentiellement sur l'approche des N-Grams et ses applications dans l'analyse textuelle et le traitement de la parole, donc un état de l'art de cette approche est présenté dans le but de faire un balayage de ce qui a été fait pour l'information multimédia texte et son. Nous expliquerons aussi l'intérêt de cette approche, les avantages et les inconvénients et les preuves des bons résultats qu'a montré cette notion pour le nettoyage des données, la catégorisation de documents multilingues, la reconnaissance automatique de la parole, l'analyse de grands corpus textuels, etc.

Nous présentons dans le dernier Chapitre les résultats de l'approche de classification des images par la technique des N-Grams. À cet égard, nous présenterons aussi le prototype SYCLIM (Système de Classification des IMages) que nous avons développé. Nous donnons de même une inscription de la base de données de nos images, ainsi que l'évaluation de nos résultats.

Nous terminerons notre mémoire par la conclusion et les perspectives de notre approche.

CHAPITRE
II

Panorama des méthodes de classification

1. Introduction

La problématique de la classification consiste à affecter les objets d'un ensemble de données à des catégories ou classes prédéfinies. Ce type de question fait partie des problèmes rencontrés lors de la phase du groupement et la classification de données. À cet égard, aucune méthode de classification n'a pu être spécifiée à un problème bien particulier. Dans ce chapitre nous présenterons un panorama des méthodes de classification les plus connues et qui font référence à l'existence de groupes ou classes de données, elles se divisent en deux groupes :

- Les méthodes de classification automatique (aussi appelées méthodes de *clustering*) : méthodes basées sur la notion d'apprentissage non supervisé, laquelle consiste à regrouper des objets appartenant à un ensemble T en classes restreintes de telle sorte que les objets d'une même classe soient le moins dispersés possible.
- Les méthodes d'affectation (aussi appelées «classificateurs») basées sur la notion d'apprentissage supervisé : méthodes utilisant un ensemble d'exemples où les classes d'appartenance sont connues au préalable. À partir de cet ensemble, des normes (ou règles) d'affectation seront définies.

Nous développerons plus loin ces différentes méthodes en soulignant certains de leurs avantages et inconvénients.

2. Méthodes de classification automatique

Les problèmes de classification automatique ont été traités à travers plusieurs ouvrages. L'objectif de ces méthodes est de regrouper les individus en un nombre restreint de classes homogènes. Dans ce type de méthodes les classes seront obtenues à l'aide d'algorithmes formalisés et non par des méthodes subjectives.

On distingue aussi les méthodes de classification non hiérarchiques et les méthodes de classification hiérarchiques.

2.1. Méthodes non hiérarchiques

Ce sont des méthodes qui produisent directement une partition en un nombre fixé de classes. Parmi ces méthodes, nous retrouvons :

2.1.1. Méthode de leader

Cette méthode considère chaque objet une seule fois. Lorsque le premier objet arrive, on lui attribue la première classe et il devient le leader de celle-ci. Ensuite, chaque fois qu'un nouvel objet se présente, on calcule sa distance par rapport aux leaders de chacune des classes existantes à cet instant, et on compare cette distance à un seuil. Si cette distance est inférieure au seuil fixé, on attribue au nouvel objet la classe du premier leader trouvé (pour lequel la distance calculée est inférieure au seuil), sinon une nouvelle classe est créée et le nouvel objet devient le leader de cette classe [SPA 80]. L'algorithme est le suivant :

1. Soit E l'ensemble à classer.
2. Soit m une dissimilarité maximale donnée.
3. Soit k=1.
4. Soit i1=1.
5. Soit i2=1.
6. Tant que i1 < |E|
   7. Tant que diss(e[i1],e[i2])<m.
      8. Ajoute e[i2] à la classe C[k].
      9. i2 := i2 + 1.
   10. FinTantQue
   11. i1 := i2.
   12. k := k +1.
13. Fin Tant que

Cette méthode dépend de l'ordre de présentation des objets. Lorsque cet ordre n'est pas optimal, le nombre de classes augmente sensiblement. Par ailleurs, pour définir des nouveaux leaders, cette méthode utilise des distances, ce qui nous ramène au problème de la définition des métriques.

2.1.2. Méthode de *k*-means

Cette méthode est encore appelée *algorithme des centres mobiles*. Dans ce type d'algorithme, la classe est représentée par son centre de gravité. L'algorithme des *k*-means mis au point par McQueen en 1967 est l'un des algorithmes de clustering les plus connus. Il est basé sur la méthode des centroïdes (ou centres de gravité). Le principe de cette méthode est le suivant :

On se donne pour commencer, *k* centres arbitraires *c1, c2,..., ck* où chaque *ci* représente le centre d'une classe C^i. Chaque classe C^i est représentée par un ensemble d'individus plus proches de *ci* que de tout autre centre. Après cette initialisation, on effectue une deuxième partition en regroupant les individus autour des mj qui prennent alors la place des *cj* (*mj* est le centre de gravité de la classe C^j, calculé en utilisant les nouvelles classes obtenues). Le processus est ainsi réitéré jusqu'à atteindre un état de stabilité où aucune amélioration n'est possible. L'algorithme est le suivant :

1. Soit E l'ensemble à classer.
2. Soit K le nombre de classes à créer.
3. Choisis K éléments G[1][0], ... G[K][0] de E.
4. Soit i := 1.
5. Tant que quelque chose change
   6. Pour k=1..K
      7.Construis la classe C[k][i]:={e | ||e-G[k][i-1]||<||e-G[k'][i-1]|| pour k'!=k, e appartenant à E}
      8. Determine G[k][i] := g(C[k][i]).
   9. Fin Pour
   10. i := i+1
11. Fin Tant que

Cette méthode est convergente et surtout avantageuse du point de vue calcul mais elle dépend essentiellement de la partition initiale. Il existe donc un risque d'obtenir une partition qui ne soit pas optimale mais seulement meilleure que la partition initiale. De plus, la définition de la classe se fait à partir de son centre, qui pourrait ne pas être un individu de l'ensemble à classer, d'où le risque d'obtenir des classes vides.

2.1.3. Méthode des nuées dynamiques

Cette méthode a été proposée par [DID 72]. Elle peut être considérée comme une généralisation de la méthode des centres mobiles. Le principe de la méthode est le suivant : on tire au hasard *k* noyaux parmi une famille de noyaux (chaque noyau contient un sous-ensemble d'individus). Puis chaque point de l'ensemble d'apprentissage est affecté au noyau dont il est plus proche. On obtient ainsi une partition en *k* classes dont on calcule les noyaux. On recommence le processus avec les nouveaux noyaux et ainsi de suite jusqu'à ce que la qualité de la partition ne s'améliore plus.

• *Algorithme*

1. Soit E l'ensemble à classer.
2. Soit f une fonction qui détermine un noyau d'une classe donnée.
3. Soit g une fonction qui détermine une classe autour d'un noyau donné // Noyau: c'est l'ensemble d'éléments qui agit comme un centre.
4. Soit W une fonction qui mesure l'homogénéité des classes d'une partition donnée et un ensemble de nœuds donné.
5. Soit K le nombre de classes à créer.
6. Choisis K noyaux dans E.
7. Tant que W n'est pas satisfaisant
   8. Utilise g pour déterminer une classe autour de chaque noyau.
   9. Utilise f pour déterminer les noyaux de ces classes.
10. Fin Tant que

Cette méthode a l'avantage de traiter rapidement de grands ensembles d'individus. Elle fournit une solution dépendant de la configuration initiale et nécessite le choix du nombre de classes. En général le nombre de classes est fixé par l'utilisateur et l'initialisation est faite par un tirage au hasard. Pour comparer l'individu avec les noyaux, cette méthode utilise des distances, ce qui a l'inconvénient d'établir des métriques.

En conclusion, les méthodes non hiérarchiques permettent de traiter rapidement de grands ensembles d'individus, mais elles supposent que le nombre des classes est fixé au départ. Si le

nombre de classes n'est pas connu ou si ce nombre ne correspond pas à la configuration véritable de l'ensemble d'individus (d'où le risque d'obtenir des partitions de valeurs douteuses), il faut presque toujours tester diverses valeurs de k, ce qui augmente le temps de calcul. C'est pourquoi, lorsque le nombre des individus n'est pas trop élevé, on préfère utiliser les méthodes hiérarchiques.

2.2. Méthodes hiérarchiques

La classification hiérarchique consiste à effectuer une suite de regroupements en classes de moins en moins fines en agrégeant à chaque étape les objets ou les groupes d'objets les plus proches. Elle fournit ainsi un ensemble de partitions de l'ensemble d'objets [CEL 89]. Cette approche utilise la notion de distance, qui permet de refléter l'homogénéité ou l'hétérogénéité des classes. Ainsi, on considère qu'un élément appartient à une classe s'il est plus proche de cette classe que de toutes les autres.

La figure II.1 est une illustration du principe des méthodes hiérarchiques. Dans cette figure, on représente la suite de partitions d'un ensemble $\{a, b, c, d, e\}$:

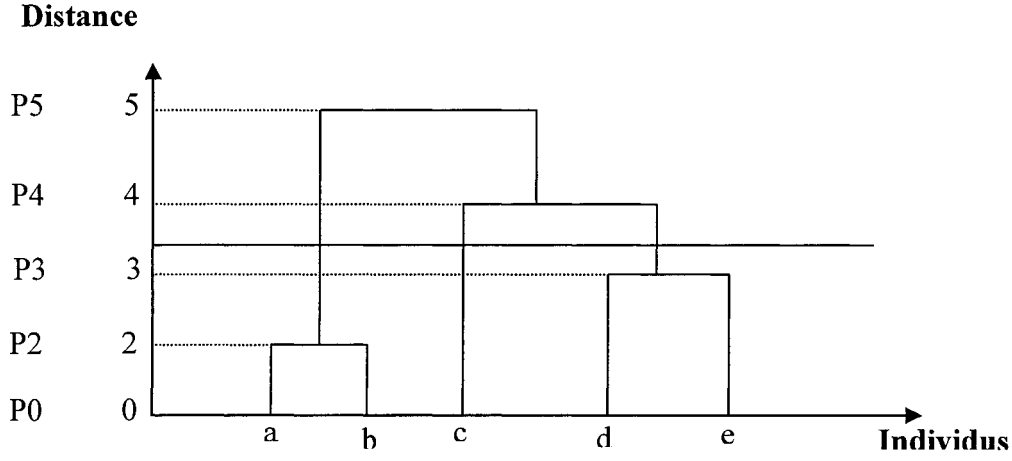


Figure II.1 : La partition hiérarchique.

Les différentes partitions représentées dans la figure II.1 sont :

P0 = $\{\{a\},\{b\},\{c\},\{d\},\{e\}\}$ correspond à la distance d = 0;

P1 = $\{\{a, b\},\{c\},\{d\},\{e\}\}$ correspond à la distance d = 2;

P2 = $\{\{a, b\},\{c\},\{d,e\}\}$ correspond à la distance d = 3;

P3 = $\{\{a, b\},\{c, d, e\}\}$ correspond à la distance d = 4;

P4 = $\{\{a, b, c, d, e\}\}$ correspond à la distance d = 5.

À chaque partition correspond une valeur numérique représentant le niveau auquel ont lieu les regroupements. Les partitions sont définies en coupant l'arbre à un certain niveau en regardant les branches qui tombent. Dans l'exemple de la figure II.1, si on coupe l'arbre à une valeur 3.5 on aura la partition suivante : P2 = $\{\{a, b\},\{c\},\{d, e\}\}$.

La principale difficulté présentée par cette méthode est la définition du critère de regroupement de deux classes, c'est-à-dire la détermination d'une distance entre les classes.

Il en existe un bon nombre d'algorithmes pour les méthodes hiérarchiques, on se limite à présenter l'algorithme de la méthode directe, elle est fondée sur un critère d'agrégation, dont la dissimilarité de classes dépend uniquement sur la dissimilarité entre objets. L'algorithme est le suivant:

1. Soit E l'ensemble à classifier.
2. Calcule et écris le tableau T de la dissimilarité *diss*.
3. Soit H l'ensemble des singletons.
4. Tant que T a plus d'une colonne.
   5. Détermine la valeur minimale m de T, soit la position C1,C2.
   6. Ajoute (C1 ∨ C2) à H avec v(C1 ∨ C2) = m.
   7. Agrège les colonnes et lignes de C1 et C2 en mettant pour toute classe C la valeur D(C, C1 ∨ C2)
      // Cela revient simplement au maximum ou au minimum des valeurs anciennes, selon le choix de D.
   8. Soit T ce nouveau tableau.
9. Fin Tant que
10. Ajoute E à H avec v(E)=valeur de T.
11. H est l'hiérarchie.

Les méthodes de classification automatique ont apporté une aide précieuse, notamment par leurs applications médicales en exploitant les informations et les données dans le domaine de la santé publique, de la recherche clinique, de l'épidémiologie, de la documentation ou de la

décision médicale. L'une des plus importantes applications de la classification automatique dans le domaine médical est la nosologie (science de la classification des maladies).

3. Méthodes d'affectation

Les méthodes d'affectation ou "classificateurs" sont caractérisées par la phase d'apprentissage qui consiste à établir des règles de classification à partir des connaissances disponibles a priori. Cette phase peut être réalisée à partir d'un apprentissage inductif ou déductif. Le premier type d'apprentissage permet de passer de cas particuliers à des lois plus générales «si les hommes *x, y, z*, etc. sont mortels, alors on peut poser comme hypothèse d'induction que l'homme est mortel». Par contre le deuxième type permet de passer d'un cas général à un cas plus particulier «si l'hypothèse que tous les hommes sont mortels, est vraie, alors en conclusion Socrate, qui est un homme, est mortel».

Les méthodes présentées dans ce chapitre utilisent soit l'apprentissage inductif soit l'apprentissage déductif mais pas les deux à la fois. Ces méthodes interviennent dans plusieurs domaines tels que la reconnaissance des formes, les statistiques, les réseaux connexionnistes (réseaux de neurones artificiels), l'intelligence artificielle. Nous aborderons ici quelques-unes de ces méthodes dans chacun de ces domaines.

3.1. Méthodes d'apprentissage inductif

Les méthodes d'apprentissage inductif consistent à inférer des règles de décision à partir d'exemples des différentes classes. Ceci se fait dans le but d'une généralisation afin de prédire des nouveaux cas, sur base des paramètres les décrivant. Parmi les méthodes utilisant ce type d'apprentissage on trouve :

3.1.1. Méthode des *k* plus proches voisins (*k*-ppv)

Le principe général de la méthode des *k*-ppv consiste à rechercher parmi l'ensemble d'apprentissage T, contenant l'ensemble des individus et leurs classes d'affectation, un nombre *k* d'individus parmi les plus proches possibles de l'individu à classer. Puis, l'individu est affecté à la classe majoritaire parmi ces *k* individus trouvés. Le nombre *k* est fixé a priori par l'utilisateur [DAS 91].

Si $k = 1$, alors l'individu est affecté à la classe du plus proche voisin de l'ensemble T. Une variante de la règle de la majorité consiste à prévoir un seuil s au-dessus duquel une décision de rejet est prise. Ainsi, on peut rencontrer des cas où l'individu n'est affecté à aucune classe. Soit l'exemple de la figure II.2 avec deux dimensions correspondant aux attributs *e1* et *e2*, et avec k=3.

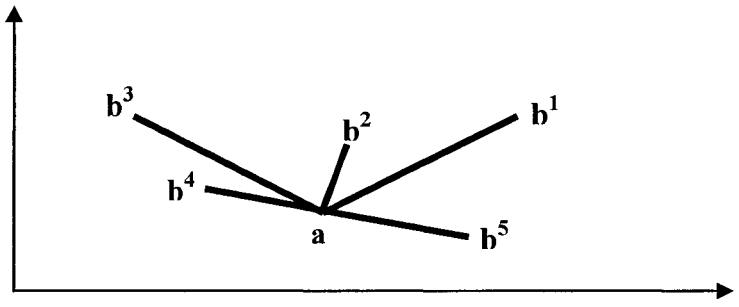


Figure II.2 : Méthode des 3-pvv.

Dans cet exemple les trois plus proches voisins de *a* sont *b4, b2* et *b5*, donc *a* sera affecté à la classe majoritaire parmi ces trois points.

• *Algorithme*

1. Entrée : x_j , j = 1,….,n +M.
2. Initialisation Dictionnaire $D_1 = \{x_1; \_ \_ \_ ; x_n\}$, m :=1
3. Tant que m =< M répéter
   4. Déterminer les k plus proches voisins de x_n+m dans D_m
   5.attribuer à x_n+m la classe majoritaire
   6.m := m + 1
7. Fin Tant que
8. Sortie : classification des x_j , j = n + 1,…..,n +M.

La méthode des k-ppv a l'avantage d'être très simple à mettre en oeuvre et d'utiliser directement l'ensemble d'apprentissage T. Elle ne fait aucune hypothèse a priori sur les données. La qualité de la discrimination par cette méthode dépend du choix du nombre k de voisins considérés. Il est cependant souvent nécessaire de faire varier ce nombre k pour obtenir les meilleurs résultats possibles. Un autre problème important de la méthode des k-ppv est qu'elle nécessite un espace mémoire très important pour stocker les données et pour faire les différents calculs dans la phase de classification. De plus, elle a l'inconvénient d'utiliser les distances pour

déterminer les voisins de l'individu à affecter, ce qui peut poser des problèmes si les dimensions à agréger ne sont pas homogènes. Afin de remédier à l'inconvénient de l'utilisation de distances, on a recours à l'utilisation des relations de ressemblances floues [PER 98].

3.1.2. Affectation par la méthode bayésienne

L'approche bayésienne a pour but de minimiser la probabilité d'erreur de classification, c'est-à-dire la probabilité jointe qu'une observation x soit en provenance d'une classe C^i et soit classée dans une autre:

• *Principe de l'algorithme*

$$P(erreur) = \sum_i \sum_{j \neq i} P(x \in C^i \, et \_ x \_ classée \_ dans \_ C^j) \qquad (1)$$

ou de façon équivalente, maximiser la probabilité de bonne classification :

$$P(correct) = \sum_i P(x \in C^i et \_ x \_ classée \_ dans \_ C^i)$$

Une caractéristique importante des données soumises à la méthode est la probabilité $P(C^i)$ avec laquelle les différentes classes apparaissent dans la population considérée. Elle est appelée *probabilité a priori*. En pratique, cette distribution est estimée à partir des fréquences observées dans les données, sauf si une connaissance a priori du domaine peut les fournir.

Considérons un vecteur x composé des valeurs des différentes variables descriptives attribuées à l'un des cas de la base de donnée. Cette information peut être cette fois utilisée pour prédire la classe du cas considéré. La règle de classification assurant une probabilité d'erreur minimum (1) est dans ce cas celle qui classe la donnée x dans la classe pour laquelle la probabilité conditionnelle de la classe étant donné x, $P(C^i/x)$, est maximum:

$$P(C^i / x) > P(C^j / x), pour \_ tout \_ j \neq i \qquad (2)$$

$P(C^i/x)$: est la probabilité conditionnelle d'appartenance à la classe C^i, sachant qu'on est au point x. Elle est appelée *probabilité a posteriori* et elle peut être calculée grâce au théorème de Bayes, sur base de la probabilité *a priori* $P(C^i)$ et de la probabilité conditionnelle $P(x/C^i)$ (distribution dans chaque classe) :

$$P(C^i / x) = \frac{P(C^i) \times P(x / C^i)}{P(x)} \qquad (3)$$

Remarquons que la classe dont la probabilité *a posteriori* est maximum peut être déterminée sans connaître $P(x)$ qui est indépendante des classes.

Pour déterminer la probabilité *a posteriori* $P(C^i/x)$ l'approche bayésienne suppose donc la connaissance des probabilités *a priori* $P(C^i)$ et de la distribution dans chaque classe $P(x/C^i)$ Celles-ci ne sont pas forcément connues, elles devront donc être évaluées à l'aide d'une méthode d'estimation. On distingue principalement les techniques paramétriques et les techniques non paramétriques. Les méthodes dites paramétriques posent des hypothèses concernant la nature des distributions (souvent supposées gaussiennes). Le problème se réduit alors à estimer les paramètres des distributions, par exemple les moyennes et les variances. Les méthodes non-paramétriques ne posent pas de telles hypothèses et procèdent par estimation de densité. Citons, parmi les méthodes les plus utilisées: les méthodes d'estimation non-paramétriques de la densité, connues également sous le nom de méthodes *des noyaux* (ou *fenêtres de parzen*) et les méthodes utilisant les k plus proches voisins [CHA 94].

3.1.3. Méthodes d'analyse discriminante

Les méthodes d'analyse discriminante ont été largement étudiées ; la littérature à ce sujet est très abondante. Nous présentons dans ce paragraphe une brève description de ces méthodes.

Le but de ces méthodes est de produire des décisions concernant l'appartenance ou non d'un objet à une classe en utilisant des fonctions discriminantes appelées également *fonctions de décisions*. Suivant les formes des classes, on peut trouver différents types de discrimination :

- *Principe et algorithme de la discrimination linéaire*

Elle consiste à séparer les classes par des frontières linéaires afin de regrouper les points à classer autour du centre de gravité de la classe (la moyenne de la classe) et à créer aussi des frontières linéaires entre les classes.

Dans le cas où on aurait n variables, la fonction de discrimination appliquée sur un objet a devient :

$$d(a) = w_1 x_1 + w_2 x_2 + ... + w_n x_n + w_{n+1} \qquad (4)$$

Cette fonction dépend de paramètres *w1, ..., wn, wn+1*. La détermination de ces paramètres se fait par un algorithme d'apprentissage qui vise à satisfaire le critère associé au modèle. En

fonction des données, le critère le plus utilisé pour ajuster ces paramètres est celui qui vise généralement à minimiser l'erreur de classification [MCL 92].

Si on a k classes, on définit k fonctions de discrimination :

$$d_i(a) = W_i X^t \text{ avec } W_i = (w_{i1}, w_{i2}, \ldots\ldots, w_{in}) \text{ et } X = (x_1, x_2, \ldots\ldots, x_n, 1) \qquad (5)$$

La règle d'affectation est donnée comme suit (figure II.3) :

$$\text{Si } \quad d_i(a) > 0 \quad \text{alors} \quad a \in C^i \quad \text{pour } i = 1, \ldots\ldots, k$$

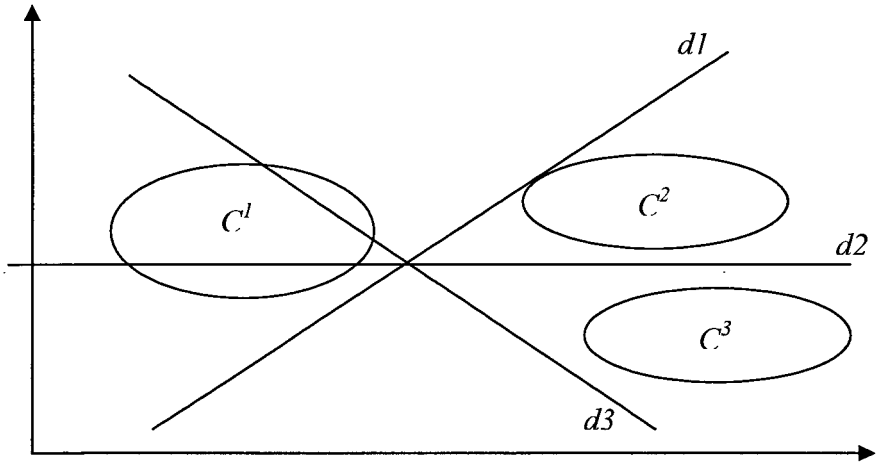


Figure II.3. Séparation linéaire pour 3 classes.

• *Principe et algorithme de la fonction discriminante quadratique*

Le principe de cette méthode est le même que celui développé précédemment excepté qu'au lieu de séparer les classes par des hyperplans, on les sépare par des surfaces qui ont généralement la forme ellipsoïde. La discrimination quadratique utilise plusieurs métriques (une par classe) pour mesurer la dispersion de chaque classe et la règle de décision est donnée comme suit :

On affecte l'objet a à la classe C^h si :

$$(x - g_h)^t M_h (x - g_h) = Min_{l=1\ldots k} (x - g_t)^t M_l (x - g_t)$$

Où x : est un vecteur composé des valeurs des différentes variables descriptives attribuées à l'objet a.

g_l = le centre de gravité de la classe C^l

M_l = la métrique de la classe C^l.

Les méthodes d'analyse discriminante ont comme difficulté le choix de la métrique à utiliser afin d'obtenir des classes où les points d'une même classe soient les moins dispersés possibles autour du centre de gravité de la classe. Ce sont des méthodes totalement compensatoires qui appliquent une agrégation globale sur les performances des attributs de l'objet. Ceci a pour conséquence un côté arbitraire de la méthode vu l'hétérogénéité des données.

L'analyse discriminante peut être utilisée dans le domaine médical par exemple, en affectant un patient à une classe diagnostic en fonction de la valeur de ses paramètres xi, $i=1,...,n$. L'ensemble d'apprentissage permet de trouver la fonction discriminante en estimant les coefficients wi. À partir de cette fonction de décision, on peut affecter n'importe quel patient. Reprenons l'exemple suivant :

Soit deux diagnostics médicaux(appendicite et salpingite) et trois signes (DEF : Défense ; DFID : douleur de la fosse iliaque droite ; DFIG : douleur de la fosse iliaque gauche). En utilisant la fonction de discrimination donnée par (4) et après l'estimation des coefficients des paramètres, on aura les règles de décisions suivantes :

$$f(\text{appendicite}) = 4.DEF + 10.DFID - 10.DFIG$$

$$f(\text{salpingite}) = 3.DEF + 5.DFID + 5.DFIG$$

Si un patient ne présente pas de signe de défense et présente les signes de douleur des fosses iliaques droite et gauche on aura :

$$f(\text{appendicite}) = 4*0 + 10*1 - 10*1 = 0$$

$$f(\text{salpingite}) = 3*0 + 5*1 + 5*1 = 10$$

D'après ce calcul, le diagnostic est en faveur d'une salpingite.

3.1.4. L'approche des réseaux de neurones

Les réseaux de neurones sont à l'origine d'une tentative de modélisation mathématique du cerveau humain. Le principe général consiste à définir des unités simples appelées neurones, chacune étant capable de réaliser quelques calculs élémentaires sur des données numériques. On relie ensuite un nombre important de ces unités formant ainsi un outil de calcul puissant.

L'étude de réseaux de neurones artificiels est très ancienne et a été étendue aux problèmes de classification et reconnaissance des formes. Commençons d'abord par donner quelques définitions relatives à la théorie des réseaux de neurones.

• *Neurone artificiel*

Un neurone est une unité de traitement de l'information. La figure II.4 en donne une représentation schématique.

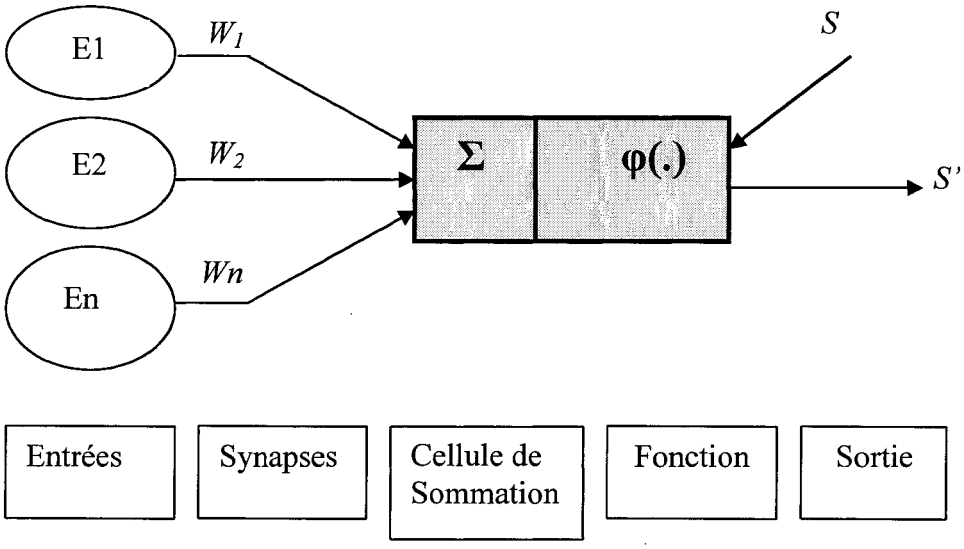


Figure II.4 : Structure d'un neurone artificiel.

Les valeurs des entrées *E1, ..., En* représentent en général les attributs d'un objet à classer et les poids *W1, ..., Wn* (ou coefficients synaptiques) associés aux entrées sont des variables de la fonction score du poids, appelée aussi *fonction d'activation du neurone* (la fonction d'activation la plus utilisée est la somme pondérée des valeurs d'entrée). La valeur d'activation est ensuite passée comme argument à la fonction de sortie qui détermine la valeur de sortie du neurone *S'*. L'entrée supplémentaire *S* sert à indiquer au neurone la valeur de sortie attendue pour qu'il puisse corriger ses coefficients synaptiques et s'approche de cette valeur.

• *Réseau de neurones*

Un réseau de neurones se compose de neurones connectés de façon à ce que la sortie d'un neurone puisse être l'entrée d'un ou plusieurs autres neurones. Les connexions entre les neurones sont dotées de poids (figure II.5) :

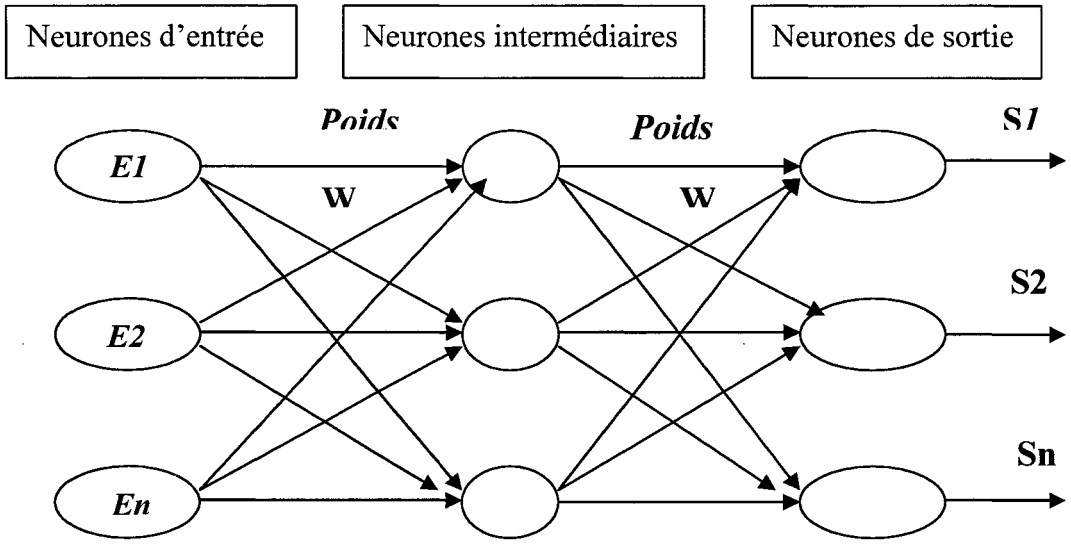


Figure II.5: Réseau de neurones artificiels.

Le principe général des méthodes utilisant les réseaux de neurones consiste à modifier (ou ajuster) les paramètres comme, par exemple, le *s* poids et les seuils par des algorithmes itératifs afin d'obtenir des réponses correctes.

L'objectif de ces algorithmes est de minimiser une mesure d'erreur. La mesure la plus utilisée est celle de l'erreur des moindres carrés, ce qui revient à minimiser l'expression :

$$E = \sum_{l=1}^{k} \sum_{i=1}^{m} (S_{il} - S'_{il})^2 \qquad (6)$$

d'où E est la variable à minimiser, *Sil* la sortie *i* attendue et *S'il* la sortie *i* du réseau pour l'exemple *l*.

Parmi les méthodes de réseaux de neurones utilisées dans le cadre des problèmes de classification nous citerons :

• *Méthode du perceptron à une seule sortie*

Cette méthode consiste à donner une décision d'appartenance ou non d'un objet à une classe (figure II.6).

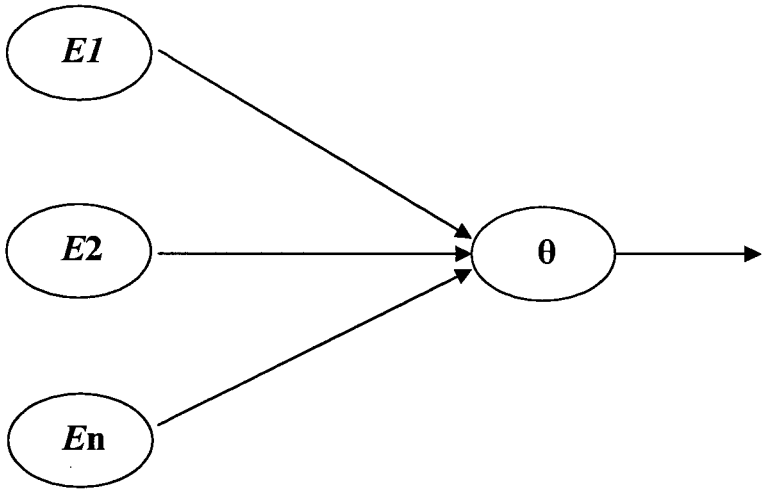


Figure II.6 : Forme générale d'un perceptron à une seule sortie (*d'adaline*).

Avec :

$$S = \begin{cases} 1 & \text{si } \sum_{i=1}^{n} w_i \times E_i + \theta > 0 \\ 0 & \text{sinon} \end{cases} \qquad (7)$$

On classe x à la classe C^1 si S = 1 et à C^2 sinon

L'équation (7) du perceptron a la même forme que la fonction de discrimination linéaire donnée par l'équation (4) utilisée dans les méthodes d'analyse discriminante, ce qui signifie que les méthodes du perceptron sont utilisées pour discriminer des individus linéairement séparables.

• *Méthode du perceptron multicouches*

Afin de traiter les problèmes de classification à plus de deux classes qui ne sont pas obligatoirement linéairement séparables, on utilise les réseaux à couches. Les réseaux à couches sont connus sous le nom de *perceptron multicouches*. Ce sont des réseaux où les neurones sont regroupés en couches connectées entre elles. On distingue trois types de couches : la couche d'entrée, la couche de sortie et les couches cachées (figure II.7).

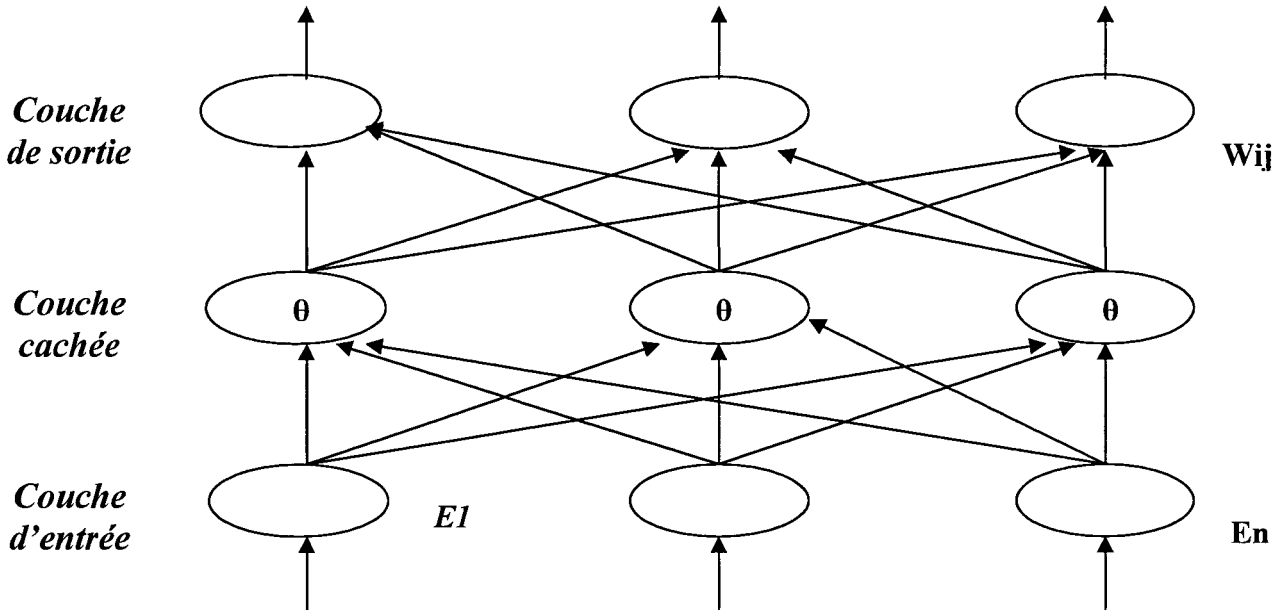
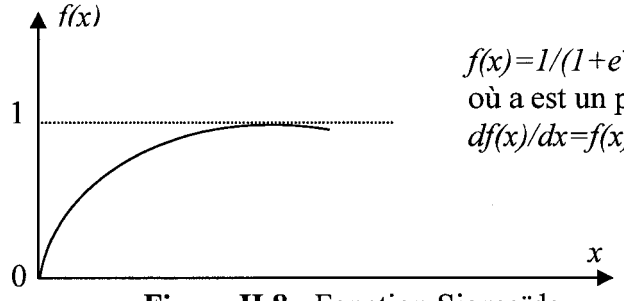


Figure II.7 : Forme générale d'un réseau avec trois couches.

Les perceptrons multicouches utilisent le principe de rétro-propagation *back-propagation* qui est une généralisation de la méthode de l'erreur des moindres carrés proposée par [RUM 86]. Le principe de la rétro-propagation est d'optimiser les paramètres du réseau de neurones en utilisant la technique de descente du gradient. On confronte le réseau à des exemples déjà classés. Lorsqu'un résultat est obtenu, l'erreur de classification est calculée (par exemple l'équation (6), permet de calculer l'erreur quadratique). Par la suite, cette erreur est rétropropagée d'une couche à l'autre en partant de la couche de sortie pour que les poids puissent être modifiés en fonction de l'erreur commise.

Dans le cas où les classes ne seraient pas linéairement séparables, la méthode de rétro-propagation ne peut pas utiliser la fonction à seuil, comme celle utilisée dans l'équation (7). Dans ce cas, elle utilise une fonction dérivable connue comme fonction sigmoïde. Parmi ces fonctions on trouve la fonction logistique (figure II.8).



$f(x)=1/(1+e^{-ax})$
où a est un paramètre de pente de f
$df(x)/dx=f(x)(1-f(x))$ et $df(0)/d(x)=a/4$

Figure II.8 : Fonction Sigmoïde.

Ce type de réseaux est le plus utilisé dans les problèmes de classification et il a fourni de bons résultats. Néanmoins, la convergence en terme d'apprentissage n'a pas été démontrée : le temps de calcul requis par l'apprentissage pour corriger les paramètres peut être très élevé et la convergence non immédiate. En outre, les réseaux de neurones produisent automatiquement la décision et sans l'intervention du décideur. Ce qui leur a attribué le nom de la boîte noire (figure II.9). Ceci permet de dire que les méthodes de réseaux de neurones sont des méthodes à caractère non explicatif et la décision prise n'est pas justifiée.

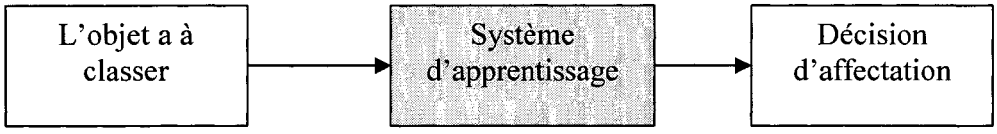


Figure II.9 : L'effet de la boîte noire.

• *Le réseau à architecture évolutive ART*

ART (Adaptive Resonance Theory) est un modèle de réseau de neurones à architecture évolutive développé en 1987 par Carpenter et Grossberg. Dans la plupart des réseaux de neurones, deux étapes sont considérées. La première est la phase d'apprentissage : les poids des connexions sont modifiés selon une règle d'apprentissage, la deuxième est la phase d'exécution où les poids ne sont plus modifiés. Avec le réseau ART, ces deux étapes sont réalisées simultanément. Le réseau en phase de test, s'adapte à des entrées inconnues en construisant de nouvelles classes (ajout de neurones) tout en dégradant au minimum les informations déja mémorisées. Il existe plusieurs versions de réseaux (ART1, ART2, ART3). Le réseau ART1 est un réseau à entrées binaires.

• *Structure*

Le réseau ART1 est formé d'une couche d'entrée qui est aussi la couche de sortie et d'une couche cachée. Le terme de couche cachée est emprunté au réseau multicouche, il souligne le fait que cette couche n'est pas directement observable par l'utilisateur à la différence de l'entrée ou de la sortie. Il n'y a pas de connexion entre les neurones d'entrées. Par contre, la couche cachée est une couche d'activation compétitive, tous les neurones sont reliés les uns aux autres par des connexions inhibitrices de poids fixes. Chaque neurone de la couche d'entrée est relié à tous les

neurones de la couche cachée et, réciproquement, chaque neurone de la couche cachée est relié à tous les neurones de la couche de sortie. À chaque connexion est associé un poids.

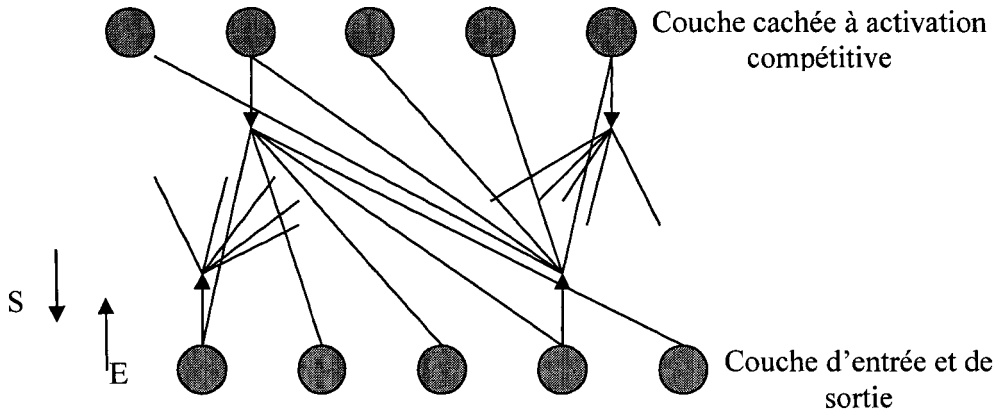


Figure II.10 : Architecture du réseau ART1.

La couche d'entrée est aussi celle de sortie. Tous les neurones de la couche d'entrée sont reliés à tous les neurones de la couche cachée et tous les neurones de la couche cachée à chacun de ceux de la couche de sortie. Il n'y a pas de relation entre les neurones d'entrée alors que la couche cachée est à activation compétitive.

• *Fonctionnement / Apprentissage*

La figure II.11 montre un vecteur d'entrée E soumis au réseau. À cette entrée correspond, après compétition entre les neurones de la couche cachée, un unique neurone j gagnant. Ce gagnant est considéré par le réseau comme le plus représentatif du vecteur d'entrée E. Le neurone j génère en retour sur la couche de sortie un vecteur S binaire (seuillage). S est ensuite comparé au vecteur d'entrée E. Si la différence est inférieure à un seuil fixé pour le réseau, le neurone gagnant est considéré comme représentant de la classe du vecteur d'entrée. Dans ce cas, la modification des poids des connexions du neurone gagnant a pour effet de consolider ses liens d'activation avec l'entrée E ; en fait l'adéquation entre ce vecteur d'entrée et cette classe est améliorée. Dans le cas contraire, le processus reprend avec les neurones de la couche cachée moins le neurone gagnant de l'étape précédente. Si tous les neurones cachés sont passés en revue sans qu'aucun ne corresponde à E, un nouveau neurone caché est ajouté, qui est initialisé comme représentant de la classe du vecteur d'entrée E.

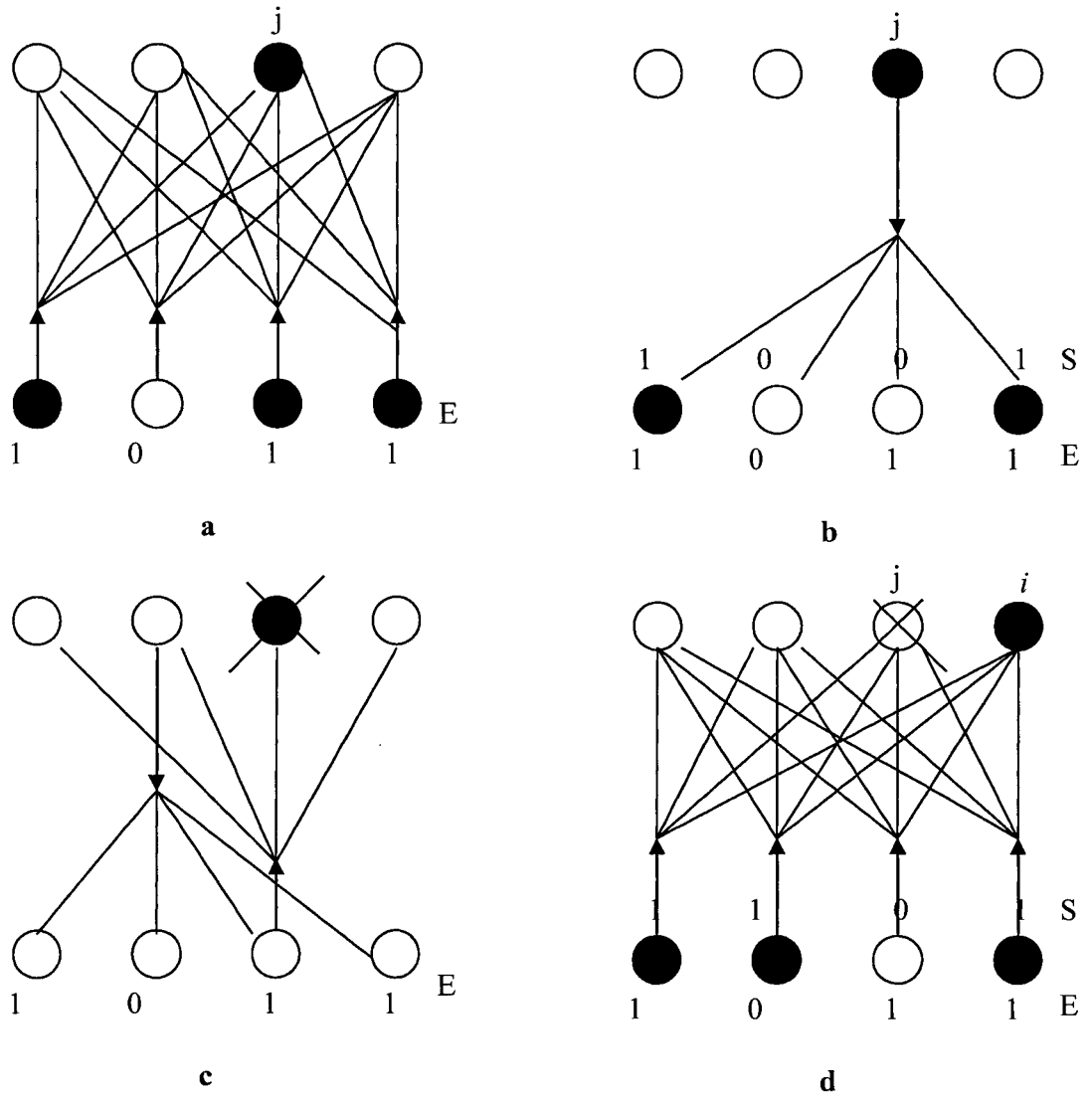


Figure II.11 : Fonctionnement du réseau ART1.

a) Présentation du vecteur d'entrée E, un neurone gagnant j est sélectionné.

b) Tentative d'unification entre S (retour du neurone j) et E.

c) Echec : suppression du neurone gagnant, présentation de E.

d) Unification : le neurone *i* est un représentant de la classe du vecteur d'entrée E.

• *Algorithme*

Ici, l'apprentissage consiste tout autant dans la détermination des poids que de la valeur du seuil d'unification ß.

1- Initialisation des poids aléatoirement entre 0 et 1 et choix d'un seuil d'unification ß.

2- Présentation d'un vecteur d'entrée El appartenant à la base d'apprentissage.

3- Calcul du neurone gagnant sur la couche cachée Nj.

4- Génération en retour d'un vecteur de sortie Sj issu de ce seul neurone Nj. Sj a été seuillé afin de le rendre binaire.

5- Tentative d'unification entre Sj et El. Soit |S| est la norme de Sj égale au nombre de composantes à 1, par exemple |(1, 0, 1, 1)| = 3.

Si |Sj| / |El| $\geq$ ß, l'unification est réalisée. Il faut modifier les poids : étape 7.

6- Sinon |Sj| / |El| < ß, le neurone gagnant Nj est inhibé.

S'il y a encore des neurones non inhibés sur la couche cachée alors retour à l'étape 3. Sinon un nouveau neurone caché est créé, initialisé comme représentant de la classe correspondant à la forme d'entrée El en utilisant la loi de modification des poids de l'étape 7.

7- Modification des poids

Couche des poids montants :

h neurones de la couche d'entrée, j neurones gagnants de la couche cachée.

Wjh = 1 / |Sj| si le neurone h est actif (valeur 1),

Wjh = 0 sinon (valeur 0).

Couche des poids descendants:

j neurones gagnants de la couche cachée, k neurones de la couche de sortie.

Wkj = 1 si le neurone k est actif,

Wkj = 0 sinon.

Retour à l'étape 2.

8- Quand le passage de tous les exemples de la base d'apprentissage n'occasionne plus aucun ajout de neurone, il faut mesurer les performances : contrôler le nombre et la qualité des classes construites. Si le nombre est trop faible, retour à l'étape 1 avec une augmentation de la valeur de ß. Si ce nombre est trop élevé, retour à l'étape 1 en diminuant la valeur de ß.

La valeur du seuil contrôle le degré d'unification recherché entre les formes à classer et les prototypes des classes. Plus la valeur du seuil est grande, meilleure est l'adéquation recherchée. La valeur du seuil doit être choisie entre 0 et 1. Le neurone i est rattaché à une classe dont le prototype générique à priori ne correspond précisément à aucune des formes de la base d'apprentissage. L'unification est réalisée lorsque le nombre d'entrées à 1 est comparable avec le nombre de retours à 1 (coactivation statistique).

• *Résultats*

Un exemple de coalescence de données issues d'une distribution parabolique est réalisé. Les coordonnées d'un ensemble de point pris sur la parabole sont soumis en données d'entrée au réseau ART1 (fig II.12a). Après quelques itérations de l'ensemble de la base d'exemple, les classes construites par le réseau sont présentées sur la figure II.12b. Ici quatre classes correspondants aux lettres a, b, c et d sont représentées, la valeur du seuil de vigilance est de 0.7. Plus la valeur de seuil est proche de 1, plus le nombre de classes créées est grand et réciproquement.

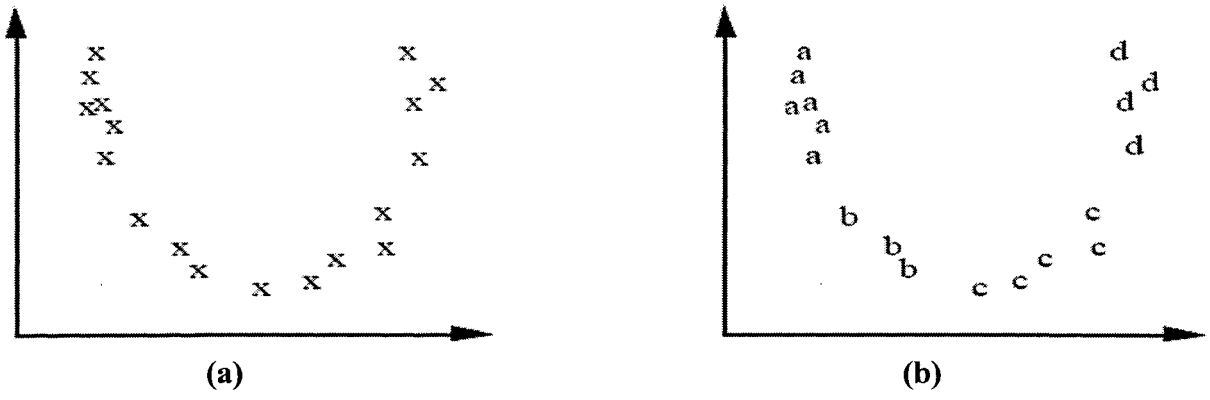


Figure II.12. Exemple de traitement réalisé par le réseau ART1.

a) Base d'apprentissage (points extraits sur une parabole).

b) Coalescence réalisée avec un seuil ß = 0.7 (4 classes).

Les applications majeures du réseau ART ont été réalisé en reconnaissance de la parole, reconnaissance des formes visuelles, en détection d'image radar ainsi qu'en classification et en coalescence.

En conclusion le réseau ART1 a une architecture en deux couches qui interagissent entre elles. Le réseau se distingue aussi par deux caractéristiques: sa flexibilité et sa stabilité pour des entrées arbitraires. Il est capable de s'adapter à des entrées non familières en construisant de nouvelles catégories ou classes (flexibilité, plasticité) mais aussi d'adapter les classes déjà apprises tout en dégradant peu les informations déjà mémorisées (stabilité). Le problème posé par ces deux notions antagonistes (flexibilité-stabilité) est résolu par le principe de l'architecture évolutive.

3.1.5. Affectation par l'approche d'arbre de décision

L'approche d'arbre de décision a été largement étudiée et appliquée dans le domaine de classification supervisée.

• *Principe*

Soit un ensemble d'individus $\{x1, x2, ..., xn\}$ que l'on veut étudier du point de vue de certaines variables ou caractéristiques fj avec $j \in J$ pour $J = \{1, ..., n\}$. Suivant les valeurs $fj(xj)$ de ces variables en xj, on est amené à effectuer telle ou telle décision sur ces individus. Par exemple:

$$Si\ (f1(x1))>39\ \ et\ \ (f2(x2))<>12/8),\ alors\ x_l \in C^l$$

Où : f1 désigne la température, f2 la tension et $x_l \in C^l$ *indique que le patient est affecté à la classe des malades.*

$(f1(x1))>39$ *et* $(f2(x2))<>12/8)$ *expriment les règles de décision alors que* $x_l \in C^l$ *représente la décision.*

Pour l'ensemble d'apprentissage $T = \{(xi, C^j) / i = 1,..., m; j=1,..., k\}$ qui contient l'ensemble des règles initiales, on souhaite obtenir un nouvel ensemble de règles qui soit aussi concis que possible. Cela peut se faire au moyen d'un arbre que nous appellerons arbre de décision.

Un arbre de décision est une structure simple récursive permettant d'exprimer un processus de classification séquentiel au cours duquel une correspondance est établie entre un objet décrit par un ensemble de caractéristiques (attributs), et un ensemble de classes disjointes. Chaque feuille de l'arbre dénote une classe et chaque nœud intérieur un test portant sur un ou plusieurs attributs, produisant un sous-arbre de décision pour chaque résultat possible du test.

• *Construction d'un arbre de décision*

Soit un ensemble d'apprentissage $T = \{(xi, Cj) / i = 1,...,m; j=1,...,k\}$. L'idée de construction d'un arbre de décision utilisant T est de raffiner T en des sous-ensembles successifs menant à des

collections d'objets comportant au plus une classe. On choisit pour cela un test portant sur un ou plusieurs résultats {R1, R2, ..., RL}.

T est ensuite partitionné en sous-ensembles T1, T2, ..., TL où Ti contient tous les individus de T présentant le résultat Ri par le test choisi. L'arbre résultant consiste en un nœud de décision identifiant le test et une branche pour chaque résultat possible. L'étape la plus importante pour la construction des arbres de décision consiste à choisir le meilleur test. Ceci permet de diminuer le plus possible le mélange des classes au sein de chaque sous-ensemble crée par le test. Ainsi, le critère de sélection le plus souvent utilisé est basé sur la théorie de l'information de Shannon. Ce type de critère est utilisé pour sélectionner les différents tests en utilisant le critère du gain d'entropie connu sous l'appellation *information mutuelle* [QUI 93]. Le processus de division des cas d'apprentissage se poursuit de manière successive jusqu'à ce que tous les sous-ensembles ne comportent plus que des individus à classe unique.

La construction par partitionnement peut conduire à des arbres extrêmement complexes qui ne permettent pas d'extraire les structures significatives des données. Afin d'obtenir un arbre plus simple et plus précis pour la classification de nouveaux cas on peut appliquer le principe d'élagage (en anglais, pruning). L'objectif de ce principe est d'améliorer les qualités de généralisation et de prédiction de l'arbre. Le principe d'élagage consiste à supprimer les parties de l'arbre jugées inutiles (ou non performante pour prédire la classe de nouveaux cas).

• *Affectation de nouveaux individus*

La règle d'affectation d'un nouvel individu s'effectue comme suit :
Partant de la racine de l'arbre, l'individu descend l'arbre jusqu'à ce qu'il arrive à une feuille. Si celle-ci représente une classe unique, il est affecté à cette classe. Si la feuille représente un mélange de classes, il est affecté à la classe majoritaire. Reprenant l'exemple suivant, soient trois classes d'Iris : Setosa (Seto), Versicolor (Vers) et Virginica (Virg). On dispose d'un échantillon de 50 individus pour chaque classe d'Iris et chaque Iris est caractérisé par quatre paramètres : longueur du sépale (Sp), largeur du sépale (SW), longueur du pétale (Pl) et largeur du pétale (Pw). L'arbre de décision est représenté dans la figure suivante :

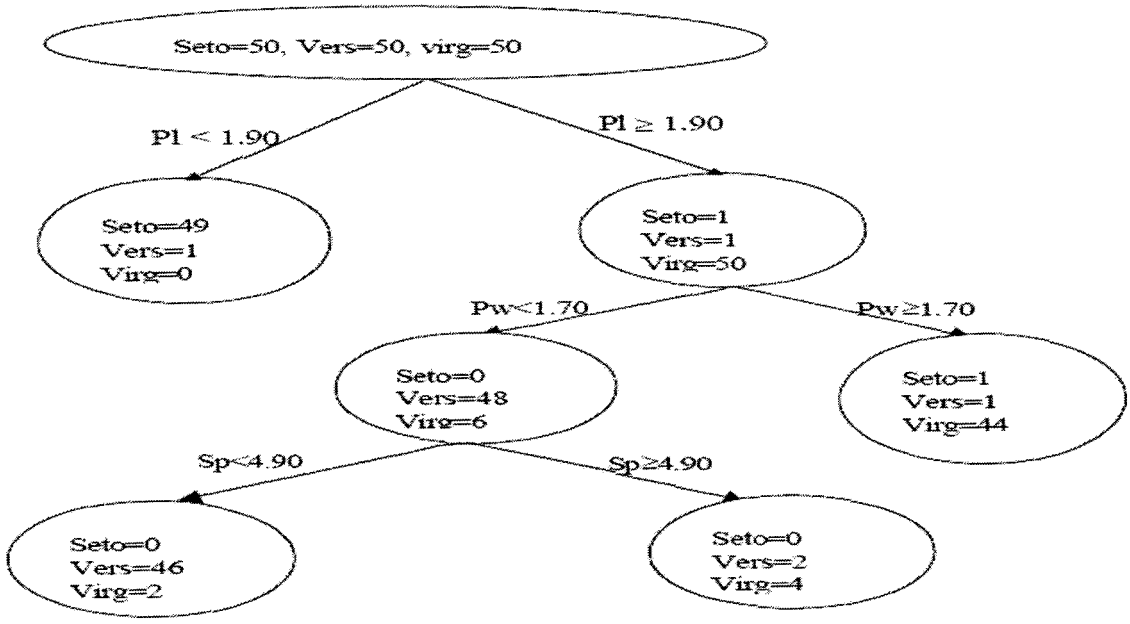


Figure II.13 : Arbre de décision sur les données d'Iris.

Dans cet exemple, sept individus sont mal classés : un Sétosa, quatre Versicolore et deux Viginica.

Cet arbre permet de fournir des règles très claires, par exemple :
Si la longueur du pétale est plus grande que 1.9 et si la largeur du pétale est plus grande que 1.7, alors l'iris est du type Virginica.

• *L'arbre de décision utilisant la probabilité bayésienne*

À chaque nœud de l'arbre, on calcule la probabilité d'appartenance à chaque classe. Ces probabilités sont conformes au théorème de Bayes sur les probabilités conditionnelles.

Considérons un échantillon de 100 personnes dont 50 en bonne santé (bs) et 50 malades m. Chaque patient est caractérisé par deux variables (ou signes cliniques) : la température et la tension. On notera P1 et P2 les probabilités *a priori* d'appartenance aux classes C1 et C 2 respectivement (avec : P1 + P2 = 1).
Dans notre exemple, nous supposerons que : P1 = P2 = 1/2. Les résultats obtenus sont présentés dans la figure II.14.

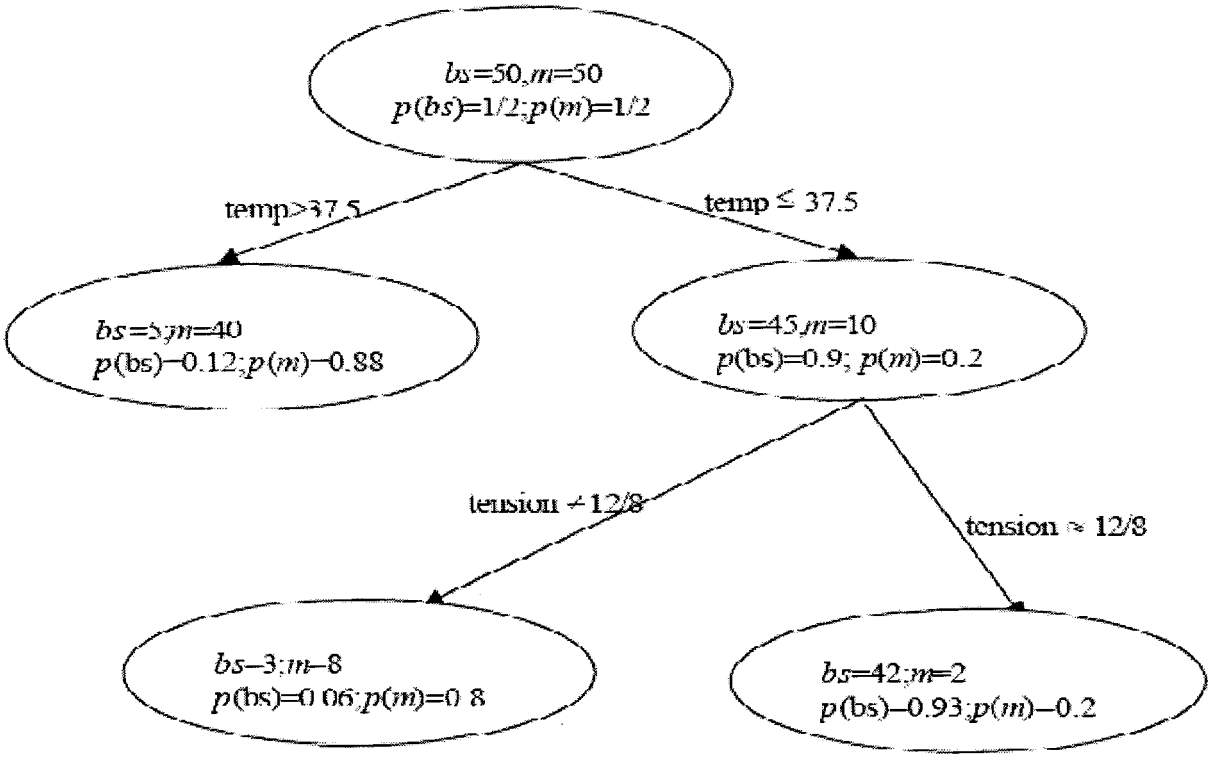


Figure II.14 : Arbre de décision utilisant les probabilités bayésienne.

La connaissance de ces probabilités ainsi que les coûts de mauvaise classification nous permettent d'évaluer les mauvaises classifications et l'espérance de coût associée à chaque feuille.

• *Algorithme d'apprentissage générique*

1. Entrée : langage de description ; échantillon *S.*
2. Initialiser à l'arbre vide ; la racine est le nœud courant.
3. Répéter
   4. Décider si le nœud courant est terminal
      5. Si le noeud est terminal alors
         6.Affecter une classe
7. Sinon
   8. Sélectionner un test et créer le sous-arbre
9. FinSi
10. Passer au nœud suivant non exploré s'il en existe Jusqu'à obtenir un arbre de décision
11. Fin Répéter

Les avantages procurés par les méthodes utilisant l'arbre de décision sont leur rapidité et, surtout, leur facilité quant à l'interprétation des règles de décision. La clarté des règles de décision rend possible le dialogue homme machine. En outre, elles ne font aucune hypothèse sur les données (méthodes non paramétriques). Par ailleurs, elles ont l'inconvénient d'être extrêmement complexes si
le nombre d'attributs et de classes augmentent.

3.2. Méthodes d'apprentissage déductif

Les méthodes d'apprentissage déductif utilisent un raisonnement analytique qui est basé sur des inférences déductives dans le but est de transformer un ensemble de connaissance sous une forme désirée par l'utilisateur. Parmi les exemples utilisant ce type d'apprentissage on a les systèmes experts et la théorie des ensembles approximatifs (rough sets). Dans ce type d'apprentissage on présentera seulement les méthodes de classification utilisant les systèmes experts.

3.2.1 Affectation par système expert

Un système expert a pour objectif de reproduire le comportement de l'expert lors de la résolution d'un problème, prenant appui sur une représentation des connaissances de ce dernier (figure II.15).

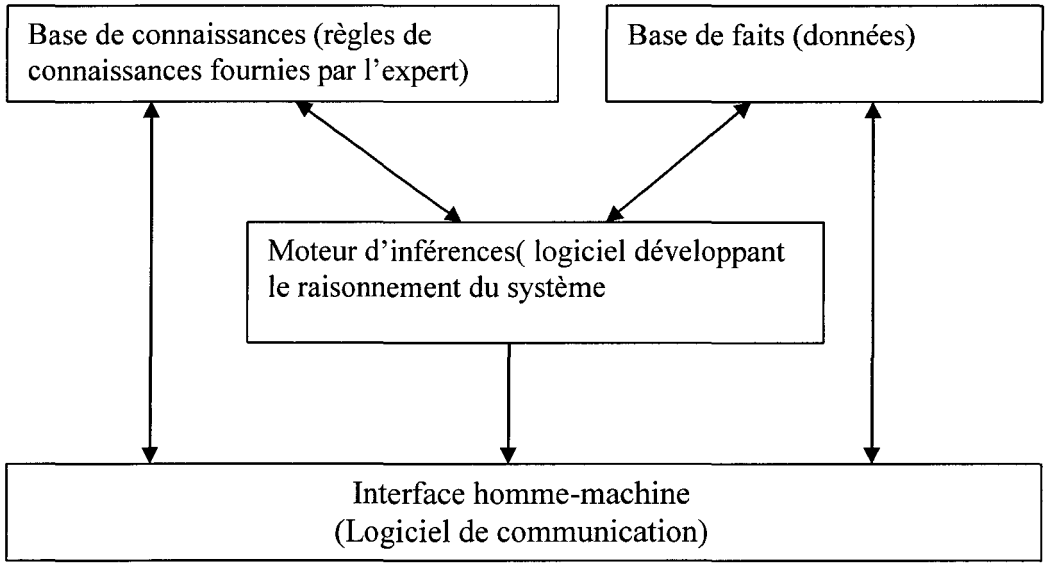


Figure II.15 : Schéma général d'un système d'expert.

Les connaissances sont représentées par une base de règles et une base de faits. Les règles sont des assertions données sous formes d'implications. Elles peuvent être interprétées comme des conditions à réaliser pour déclencher une action donnée, et elles ont la forme suivante :
Si < conditions >, alors < actions/conclusion >.

La base des faits contient des assertions qui ne sont pas exprimées sous formes d'implications. Elle représente une connaissance relevant du cas particulier de l'individu à traiter, laquelle peut être fournie au système ou bien déduite par celui-ci.

Pour affecter les individus aux différentes classes, le système cherche l'ensemble de règles applicables en effectuant un choix puis il applique la règle choisie et recommence le cycle. Le processus s'arrête lorsqu'il n'y a plus de règle applicable ou lorsque le but est atteint. Ce traitement est appelé moteur d'inférence. L'efficacité de ce raisonnement dépend de la pertinence du choix des règles.

L'affectation des individus se fait à l'aide d'un ensemble de règles comme dans les méthodes utilisant l'arbre de décision. Dans les systèmes experts, les classes et les règles d'affectation sont données par une expertise et non pas par un ensemble d'exemples (à l'inverse, les arbres de décision utilisent un ensemble d'exemples pour déterminer les classes et les règles d'affectation).

Prenons l'exemple d'un ensemble de connaissances permettant de classer quelques figures géométriques :

Règle 1 : Si figure et segments	**Alors** polygone
Règle 2 : Si figure et non segments	**Alors** ovale
Règle 3 : Si ovale et diamètre constant	**Alors** cercle
Règle 4 : Si ovale et diamètre variable	**Alors** ellipse
Règle 5 : Si polygone et trois côtés	**Alors** triangle
Règle 6 : Si polygone et quatre côtés	**Alors** quadrilatère
Règle 7 : Si quadrilatère et quatre côtés égaux	**Alors** losange

Règle 8: Si quadrilatère et côtés deux à deux parallèles **Alors** parallélogramme

Règle 9 : Si quadrilatère et côtés deux à deux égaux et angles droits **Alors** rectangle

L'un des avantages des systèmes experts est qu'il est très facile d'exprimer des connaissances certaines et précises. Leur principal inconvénient réside dans la difficulté de l'expert à exprimer sa connaissance et du grand nombre de règles nécessaires pour la plupart des applications d'intérêt pratique. En outre, les systèmes experts sont adaptés pour traiter des problèmes utilisant uniquement des variables qualitatives et à des problèmes pour lesquels il n'existe pas de solution algorithmique.

4. Performance des méthodes de classification

La plupart des méthodes de classification mentionnées dans ce chapitre ont été largement appliquées dans plusieurs domaines y compris les problèmes de classification des images. La question qu'on peut se poser est la suivante : comment évaluer les performances d'une méthode de classification ? En général, on divise l'ensemble de données disponibles en deux sous-ensembles : l'un servira pour l'apprentissage et l'autre pour le test. L'ensemble d'apprentissage est utilisé pour déterminer les paramètres du modèle de classification, par exemple les poids dans le cas d'un réseau de neurones ou les prototypes des catégories dans le cas des méthodes de classification multicritère. L'ensemble de test sert pour tester les performances de la méthode en calculant le taux de classification correcte de l'ensemble des cas. Ce taux est déterminé en divisant le nombre de cas bien classés sur le nombre des cas testés.

Parfois on est confronté à des problèmes où l'ensemble de données est restreint et on veut exploiter ces données disponibles pour construire le classificateur d'une part et tester les performances de la méthode d'autre part. Pour cela on fait appel aux techniques de rééchantillonage (resampling techniques) ; parmi lesquelles la technique de validation croisée (cross-validation) est la plus utilisée. Le principe de cette technique consiste à diviser aléatoirement l'ensemble des données en m partitions mutuellement exclusives (m-fold cross-validation). Ensuite la méthode est construite à partir de l'ensemble des partitions moins une qui servira de test. Après on réitère le processus en introduisant la partition testée dans l'ensemble d'apprentissage et en prenant une autre partition d'apprentissage pour tester la méthode et ainsi

de suite jusqu'à ce que toutes les données seront utilisées tantôt pour l'apprentissage et tantôt pour le test. La moyenne des taux de classification correcte sur toutes les partitions de test correspond au taux de prédiction.

Une autre technique de rééchantillonage déduite de *m*-fold cross-validation, appelée *leave-one-out* a été décrite par Weiss en 1991. Dans cette technique, chaque partition de test est composée d'un seul cas, tandis que tous les autres cas sont utilisés pour l'apprentissage. Ainsi, la moyenne des taux de classification correcte est déterminée en *n* itérations, où *n* représente le nombre de données disponibles.

CHAPITRE
III

Traitement d'images : État de l'art

1. Introduction

Le domaine de la vision artificielle et du traitement d'image est cette discipline qui consiste à convertir une image en données objets ou, plus explicitement, à identifier les objets contenus dans l'image par l'extraction et l'analyse de caractéristiques abstraites (*features*) à partir des pixels, suivant un processus de reconnaissance de forme similaire à celui opéré par l'humain. Ce domaine en terme générique englobe plusieurs aspects de l'imagerie tel que :

- la compression d'images
- Restauration et rehaussement
- Segmentation
- Recalage
- Fusion d'images
- Recherche d'images par contenu
- Etc....

2. Différents types d'images

2. 1 Images en noir et blanc (monochromes)

Ces images sont dites à niveaux de gris , car on ne prend pas en compte ici la couleur mais seulement l'intensité lumineuse (l'exemple classique correspond aux photographies noir et blanc). Parmi ces images on peut trouver:

- *Images binaires*

où chaque pixel est représenté par un bit (0/1) avec en général (0 pour le noir , intensité nulle et 1 pour le blanc , intensité maximale). Notons que la plupart des systèmes de traitement d'images

placeront chaque pixel dans un octet (code 0 ou 255 (pour coder le 1 de l'image binaire)) pour des facilités d'accès et d'écriture des algorithmes.

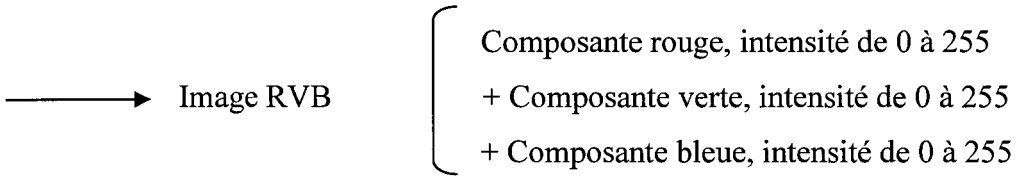
- *Images au niveaux de gris*

Dans ce cas on dispose d'une échelle de teintes de gris , et la plupart du temps on dispose de 256 niveaux de gris avec: 0 pour le noir , ...,127 pour le gris moyen , ..., 255 pour le blanc, ceci est commode car l'unité d'information est l'octet.

Certaines images peuvent être codées sur deux octets ou plus (certaines images médicales , des images astronomiques,...) ce qui peut poser des problèmes dans la mesure où les systèmes de traitement d'images courants supposent utiliser des pixels d'un octet.

2.2 Images en couleur

Ces images sont en général codées en utilisant le codage des trois couleurs fondamentales (rouge , vert , bleu) , on parle alors d'images RVB(RGB en anglais). (Cela correspond au codage des téléviseurs Français). Chaque couleur est codée sous forme d'un octet (256 couleurs). L'affichage est réalisé après passage dans une table de couleurs (transcodage).



Le codage peut ainsi être réalisé en affectant 3 bits au rouge et au vert et 2 bits au bleu (pour tenir compte par exemple de la plus faible sensibilité de la vision humaine au bleu). Il existe d'autres techniques de représentation de la couleur pour les images (on passe d'un espace 3D , l'espace RVB , à un autre espace 3D défini par une autre base).

2.3 Images 3D

Les images 3D sont des images qui représentent une scène en trois dimensions. Le « pixel » est alors appelé un *voxel*, et représente un volume élémentaire. Ces images peuvent évidemment être d'un des deux types définis précédemment (N/B ou couleur). Des exemples d'images de ce type se rencontrent dans les images médicales. Les images tomographiques

axiales sont ainsi des images construites à partir de plusieurs radiographies faites sous des angles de vue différents. Autres exemples : les images scanner , les images de résonance magnétique...

Toutes ces images pouvant être éventuellement « fusionnées » pour former des images plus complexes (les rayons X fournissant l'ossature, le scanner les tissus, et la RMN décrivant les fonctions physiologiques par exemple).

- *Cas particuliers :*

Les images obtenues par un laser tournant pour obtenir une partie de l'information de forme des objets d'une scène.

- *Images stéréographiques:*

On dispose alors d'une paire d'images (N/B ou couleur) prises sous des points de vue différents. À partir de telles images il est possible d'obtenir de l'information sur la scène 3D.

3. Filtrage des images

Les images brutes permettent rarement de parvenir à une extraction directe des objets à analyser :

- soit parce que l'éclairement de l'objet n'est pas uniforme,
- soit parce que l'objet est perçu à travers un bruit assez important : les images contiennent donc un signal et du bruit (dont on veut éliminer la plus grande partie possible),
- soit encore parce que le contraste n'est pas suffisant.

Avant d'extraire les objets et d'analyser une image, il est donc souvent nécessaire d'améliorer l'image. Il existe un grand nombre de filtres possibles, et à quelques exceptions près, on peut les classer en 2 grandes catégories : les filtres linéaires et les filtres non linéaires. Dans les filtres linéaires, la méthode de filtrage par moyenne est la plus utilisée à cause de la simplicité de son algorithme et la qualité de résultats qu'elle donne par rapport à d'autres filtres. Le cas des filtres non linéaires, c'est le filtrage médian.

3.1 Filtrage par la moyenne

Cette méthode permet de « lisser » les images, c'est à dire de diminuer les différences de niveaux de gris entre pixels voisins. Cette méthode très simple est censée supprimer le bruit. Le filtrage par la moyenne consiste à remplacer chaque pixel par la valeur moyenne de ses voisins (le pixel lui même y compris) (Tableau III.1). Cette méthode a pour effet de modifier les niveaux de gris trop différents de leurs voisins (en ce sens on peut penser « supprimer » le bruit, c'est à dire des niveaux de gris « anormaux »). Suivant la " violence" du lissage que l'on veut réaliser on choisira une taille de filtre plus ou moins grande (3x3, 5x5,..) mais on doit comprendre que les contours de l'image de départ deviendront alors plus « flous ». Les figures III.1 et III.2 illustrent l'opération de filtrage par la moyenne.

Filtre 3*3

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

Filtre 5*5

1/25	1/25	1/25	1/25	1/25
1/25	1/25	1/25	1/25	1/25
1/25	1/25	1/25	1/25	1/25
1/25	1/25	1/25	1/25	1/25
1/25	1/25	1/25	1/25	1/25

Tableau III.1 : Filtrage par la moyenne.

Image initiale

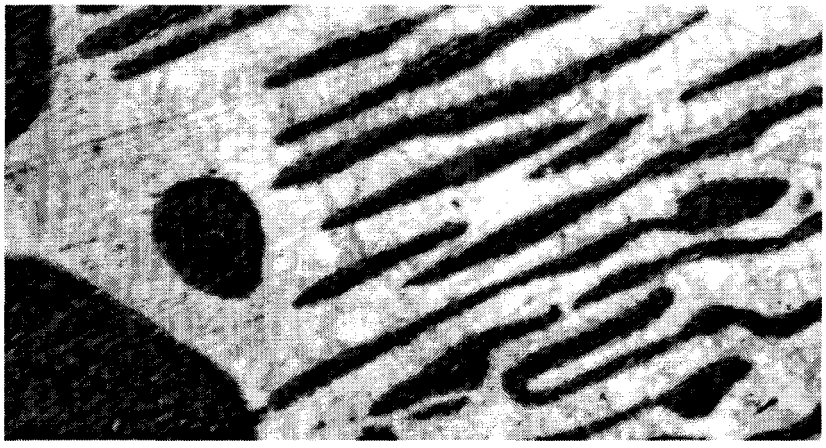


Figure III.1 : Image avant filtrage.

Image filtrée (moyenne 3X3)

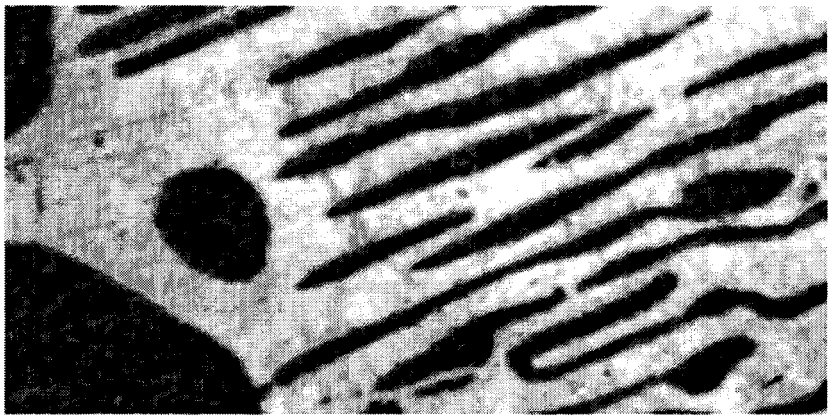


Figure III.2 : Image après filtrage.

Les inconvénients évidents de ce filtre de moyenne sont les suivants:

- Un pixel isolé avec un niveau de gris "anormal" pour son voisinage va perturber les valeurs moyennes des pixels de son voisinage.
- Sur une frontière de régions le filtre va estomper le contour et le rendre flou, ce qui est gênant en visualisation bien sûr mais éventuellement aussi pour un traitement ultérieur qui nécessiterait des frontières nettes.

Il est possible de moduler ces effets néfastes en réalisant en chaque pixel une convolution "conditionnelle". Par exemple en un pixel de niveau de gris NG1 on applique le filtre.
Supposons obtenir une valeur NG2, alors on décidera d'appliquer le filtre que si

$$|NG1 - NG2| \geq Seuil$$

3.2 Filtrage médian

Le filtre médian réalise un lissage de l'image un peu plus performant que le filtre moyenne en ce qui concerne les détails dans l'image.

- Méthode

Chaque pixel est traité en considérant ses voisins sur un voisinage donné. Le pixel lui même et ses voisins forment alors un ensemble dont on calcule la « médiane ». Le pixel sera alors remplacé par cette valeur médiane.

Exemple (Tableau III.2):

Voisinage 3*3

12	14	5	4	2
4	99	23	56	4
5	126	121	120	5
4	97	12	9	4
2	4	5	4	2

Tri 9 12 23 56 97 99 120 121 126
Valeur médiane 97

Tableau III.2 : Filtrage médian.

Intérêt du filtre médian:

- Un pixel non représentatif dans le voisinage affectera peu la valeur médiane.
- La valeur médiane choisie étant le niveau de gris d'un des pixels considérés , on ne crée pas alors de nouveaux niveaux de gris dans l'image. Ainsi lorsque le filtre passe sur un contour très marqué il le préservera mieux .

4. Segmentation d'images

Il s'agit d'une étape importante dans l'analyse d'une image. La segmentation va consister à regrouper les pixels de l'image en régions (composantes connexes). Ces régions vérifiant un critère d'homogénéité (par exemple sur les niveaux de gris ou sur la texture...). On cherche par ce traitement à obtenir une description compactée de l'image en régions. Le traitement suivant consistera probablement à mesurer la forme des régions, certaines de leurs caractéristiques et d'autres part les relations spatiales entre régions par exemple (figure III.3).

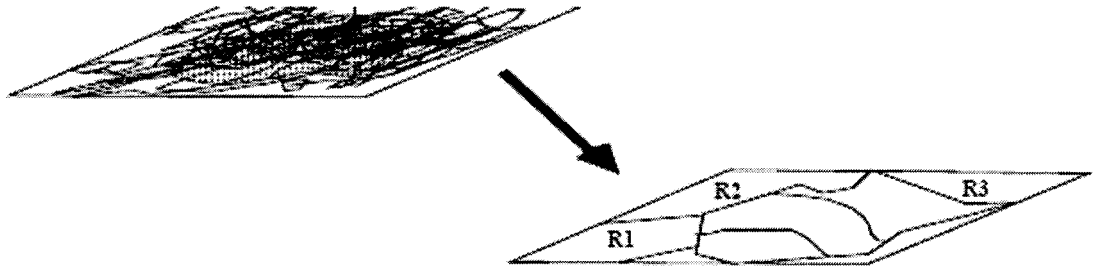


Figure III.3 : Segmentation d'une image

On voit que la segmentation peut être abordée de deux points de vue dans la mesure où une région peut être définie par l'ensemble des pixels la composant (approche région de la segmentation) ou bien par les contours de la région approche contour de la segmentation).

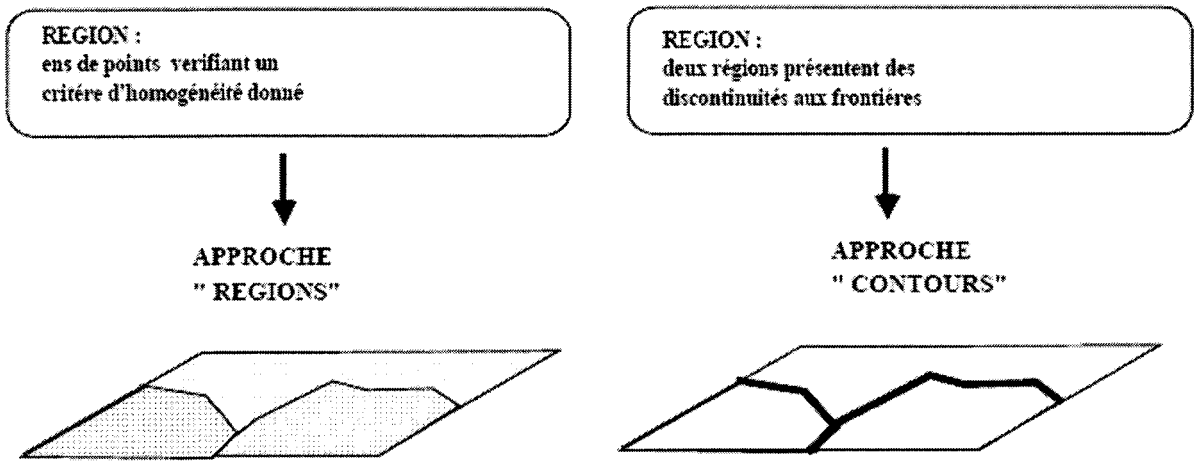


Figure III.4 : Segmentation, approche régions et l'approche contour

4.1 Segmentation : Approche par régions

Dans cette approche, on distingue deux méthodes les plus répandues :

- La première consiste à une segmentation en région par seuillage, soit en utilisant l'algorithme par recherche de sommets (de vallées) ou l'algorithme de détection de seuil par segmentation de l'histogramme.
- La deuxième est une segmentation par croissance de régions.

Pour la première méthode on va se limiter à donner une explication pour l'algorithme de détection de seuil par segmentation de l'histogramme.

4.1.1 Détection de seuil par segmentation de l'histogramme

Cette méthode développée par [OTS 79] ne s'applique que dans le cas de la segmentation d'image en deux catégories (le fonds et les objets). On suppose être dans le cas d'un histogramme où les deux populations de pixels se recouvrent partiellement. L'idée va être alors de chercher un seuil permettant d'obtenir les deux populations en minimisant une fonction de

coût. On part du point de vue que l'histogramme de l'image est en fait la somme de deux histogrammes (celui des points du fonds et celui des autres points), (figure III.5) :

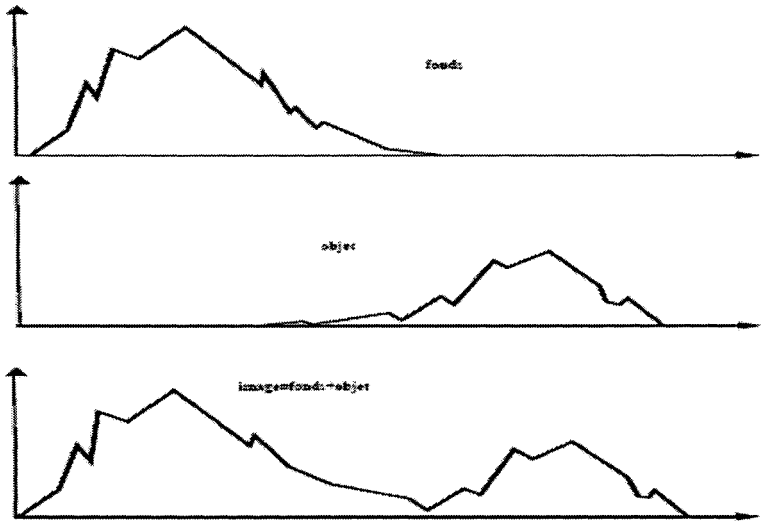
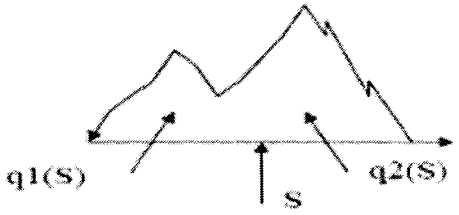


Figure III.5 : Segmentation de l'histogramme

On va alors essayer diverses valeurs de seuil et choisir celui qui sépare l'histogramme de façon optimale en deux segments (qui maximise la variance intersegments ou bien qui minimise une mesure de variance intrasegment. Ce qui peut s'exprimer ainsi:

En supposant que le nombre de niveaux de gris est 256 et que l'histogramme est noté h(i) (nombre de pixels ayant le niveau de gris i).Alors on peut définir une mesure de variance intrasegment par:

$$\sigma^2_{intra}(S) = q_1(S) \times \sigma^2_1(S) + q_2(S) \times \sigma^2_2(S)$$



Avec

$$q_1(S) = \sum_{i=0}^{S-1} h(i) \qquad \text{et} \qquad q_2(S) = \sum_{i=S}^{255} h(i)$$

$\sigma^2_1(S)$: Variance des pixels dont le niveau de gris est strictement inférieure à S

$\sigma^2_2(S)$: Variance des pixels dont le niveau de gris est supérieure ou égal à S

$$\sigma^2_{intra} = \sum_{i=0}^{S-1} h(i) \times (i-\mu_1)^2 + \sum_{i=S}^{255} (i-\mu_1)^2$$

On peut alors essayer toutes les valeurs du seuil S possibles et on garde celui qui rend $\sigma^2_{intra}(S)$ minimum. On peut aussi chercher une mesure de la « variance intersegments » en fonction du seuil S. On cherchera alors le seuil qui maximise cette mesure (c'est à dire le seuil qui sépare le « mieux » les deux segments), (figure III.6)

Une mesure possible est la suivante :

$$\sigma^2_{inter} = \sigma^2 - \sigma^2_{intra}$$

Où σ^2 est la variance globale, ce qui donne

$$q_1(S)\left[(\mu_1(S)-\mu)^2\right] + q_2(S)\left[(\mu_2(S)-\mu)^2\right]$$

Soit

$$q_1(S)\times q_2(S)\times\left[\mu_1(S)-\mu_2(S)\right]^2$$

Avec

$$\mu_1(S) = \frac{1}{q_1(S)}\sum_{i=0}^{S-1} h(i)\times i \qquad \text{et} \qquad \mu_2(S) = \frac{1}{q_2(S)}\sum_{i=S}^{255} h(i)\times i$$

On obtient alors la méthode suivante:

- Choisir un seuil S
- Calculer la moyenne de chaque segment sur l'histogramme

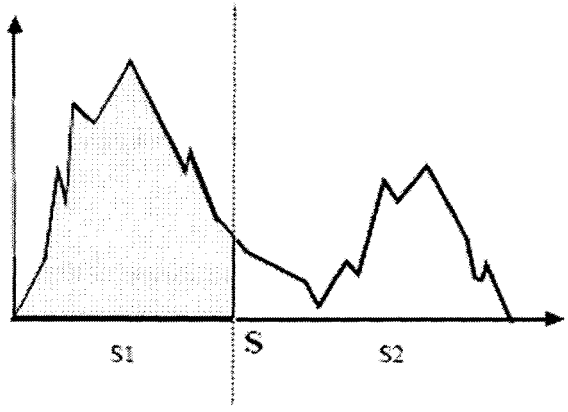


Figure III.6 : Segmentation par détection de seuil.

- Calculer la différence des moyennes au carré.
- Multiplier par le produit du nombre de pixels du segment S1 et du nombre de pixels du segment S2 et chercher le seuil qui minimise cette valeur.

On peut optimiser le calcul car quand on passe d'un seuil S au suivant S+1 on n'est pas obligé de refaire tous les calculs , on met à jour q1 et q2 de même que les moyennes.

- Recherche du seuil S par une méthode paramétrique:

Dans cette approche on part du même point de vue que précédemment, c'est à dire que l'objet et le fond sont considérés comme deux populations ayant deux distributions des niveaux de gris différentes. Mais ici on ajoute l'hypothèse supplémentaire suivante:

Chacune de ces distributions va être considérée comme Gaussienne

$$N(\mu_1,\sigma_1) \quad N(\mu_2,\sigma_2)$$

On procède alors de la manière suivante:

On essaye plusieurs seuils S et pour chacun des segments obtenus sur l'histogramme on calcule la moyenne et l'écart type et on considère qu'ils ont chacun des distributions gaussiennes. On fait alors la « somme » de ces deux distributions

$$h_S(i) = \frac{q_1(S)}{\sigma_1(S)\sqrt{2\pi}} \times e^{\frac{(i-\mu_1(S))^2}{2\times\sigma 1(S)^2}} + \frac{q_2(S)}{\sigma_2(S)\sqrt{2\pi}} \times e^{\frac{(i-\mu_2(S))^2}{2\times\sigma 2(S)^2}}$$

et on estime la « différence » avec l'histogramme sous forme d'erreur quadratique moyenne :

$$e_S = \sum_{i=0}^{255} (h(i) - h_S(i))^2$$

Le seuil S qui minimise cette erreur est considéré comme le bon seuil.

4..2 Segmentation par croissance de régions

L'idée de base est la suivante : on suppose disposer de points ou régions « amorces » et on va agréger à ces régions les pixels non encore affectés à une région. Les points clé de cette approche sont les suivants:

- Choix des points ou des régions amorce (a priori c'est le « cœur des régions » , les pixels dont on est totalement sûrs!).

- règle d'agrégation des points voisins des amorces. Prenons l'exemple suivant :

0	0	5	6	7
1	1	5	7	8
0	(1)	6	[7]	7
2	0	7	6	6
0	1	5	6	5

◯ : Amorce région 1

□ : Amorce région 2

Tableau III.3: Amorçage de régions.

Prenons maintenant comme règle d'agrégation que l'on va agréger un point à une amorce si la différence de niveau de gris est inférieure à un seuil S. Et on suppose traiter d'abord la région1 jusqu'à blocage (règle non applicable)

exemple avec S=3

(0)	(0)	[5]	[6]	[7]
(1)	(1)	[5]	[7]	[8]
(0)	(1)	[6]	[7]	[7]
(2)	(0)	[7]	[6]	[6]
(0)	(1)	[5]	[6]	[5]

(a)

Avec S=8

(0)	(0)	(5)	(6)	(7)
(1)	(1)	(5)	(7)	(8)
(0)	(1)	(6)	(7)	(7)
(2)	(0)	(7)	(6)	(6)
(0)	(1)	(5)	(6)	(5)

(b)

Tableau III.4: Amorçage de régions, (a) seuil S=3 ; (b) seuil S=8.

Exemple de technique:

- Calculer l'histogramme et repérer les modes significatifs.
- Utiliser les pixels de niveaux de gris voisins d'un mode comme amorces.

- Possibilité de travailler sur un vecteur de paramètres (niveaux de gris, couleur, texture...).
- Possibilité de définir une règle d'agrégation prenant en compte l'évolution lors de la construction de la région (par exemple différence entre le niveau de gris du point voisin et la moyenne des niveaux de gris de la région à l'étape courante).
- Possibilité si l'on connaît le type et la forme, ...etc de la région cherchée de définir un critère d'arrêt sur l'agrégation.

• *Méthodes d'Agrégation division (split and merge)*

L'idée ici consiste à étendre la méthode précédente et à diviser initialement l'image en un ensemble de régions disjointes (par exemple 4 quadrants si l'image est carrée puis ensuite de diviser ou agréger les régions suivant que des critères de division ou d'agrégation sont vérifiés ou non.

Appelons R l'image entière et R1 R2 R3 R4 .... les régions obtenues par divisions successives . Si l'on part de l'image R et que l'on divise jusqu'aux pixels on construit ce que l'on appelle une représentation de l'image en quadtree: (figure III.7)

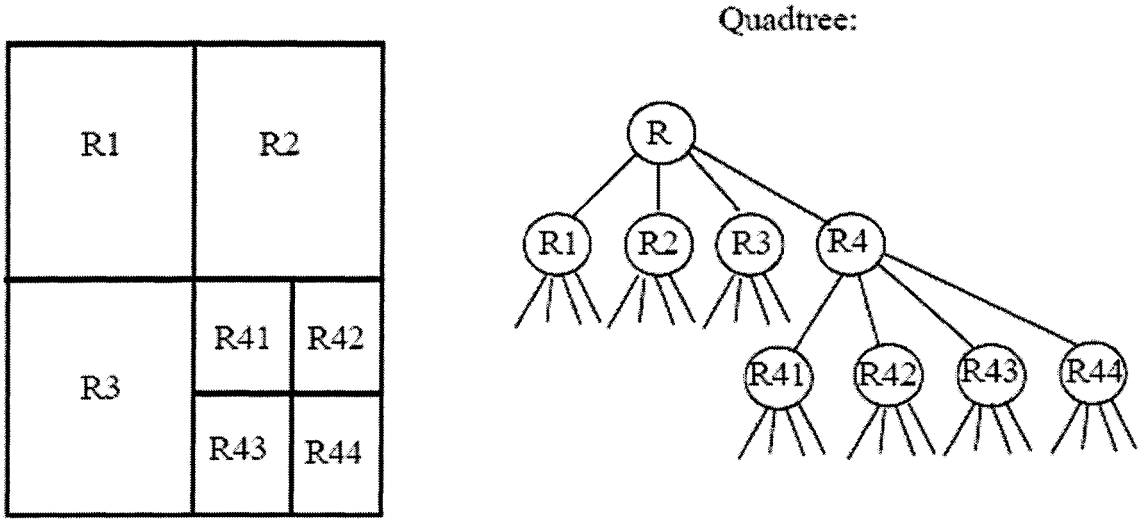


Figure III.7 : Agrégation et division d'une image.

Soit P(Ri) un prédicat logique sur une région Ri donnant comme résultat :
- Vrai si la région satisfait un critère donné d'homogénéité (par exemple tous les pixels ont le même niveau de gris (difficile à satisfaire!).

- Faux si la région ne satisfait pas ce critère. La méthode de « split and merge » peut alors s'écrire:

1. Diviser en quatre quadrants disjoints toute région Ri, où l'on a P(Ri)=Faux
2. Fusionner toutes les régions adjacentes Rj et Rk pour lesquelles on a P (Rj U Rk)= Vrai
3. Arrêter quand on ne peut plus ni fusionner ni diviser. Sinon aller en 1.

La figure III.8 illustre l'exemple du découpage d'une image R en quadrant :

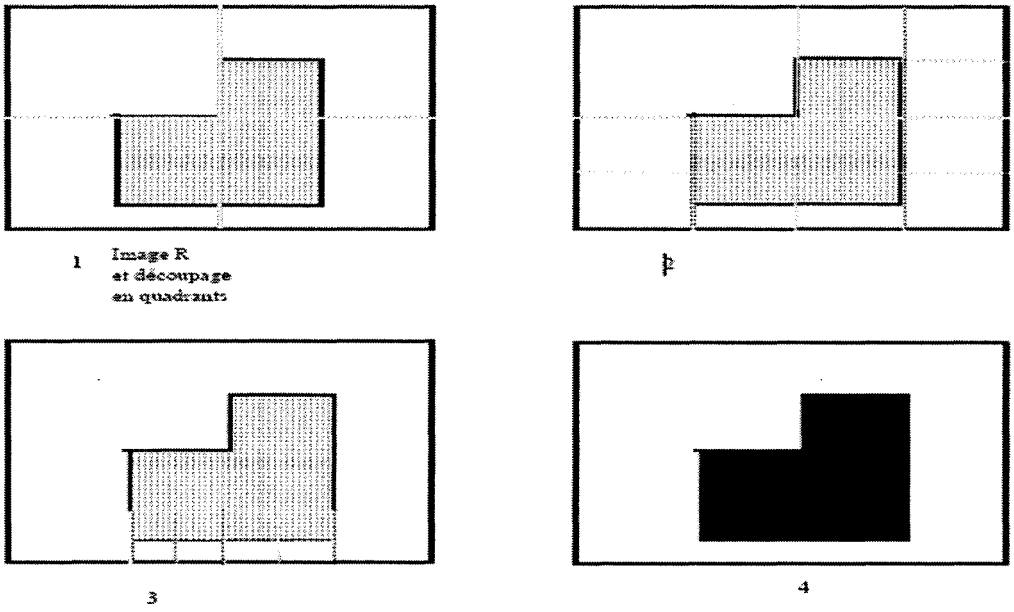


Figure III.8 : Découpage en quadtree.

5. Codage des contours et régions

On suppose ici avoir segmenter l'image et donc avoir identifier les régions et leurs contours. Le problème posé est alors la représentation des objets. On peut alors représenter les régions en mémorisant la liste des points composant la région, mais c'est lourd ! ou bien la liste des points du contour. Il existe cependant des méthodes de codage des contours et nous allons donner quelques techniques de codage de ces contours (donc des régions).

5.1 Représentation par des chaînes de codes

Dans cette méthode les contours sont représentés par une séquence connexe de segments de droite de longueur donnée et de direction donnée.

Cette représentation est basée sur une 4-connexité ou une 8-connexité des segments. Chaque segment est codé suivant le schéma suivant (figure III.9 codage de Freeman) :

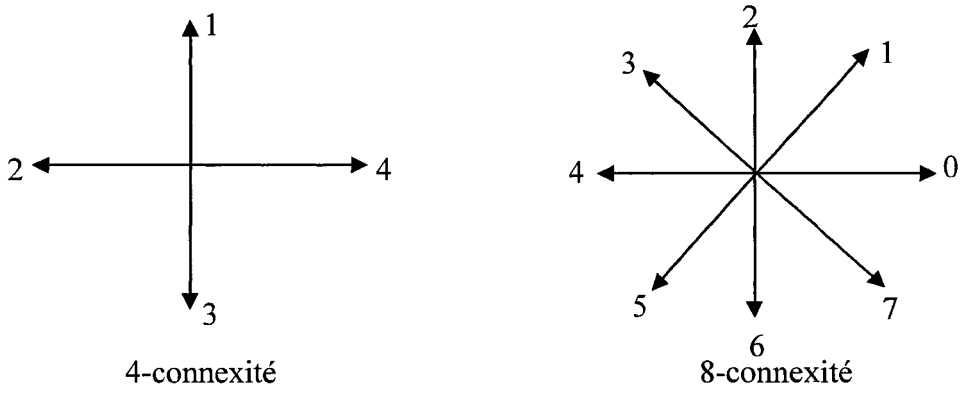


Figure III.9 : Codage de Freeman des segments.

Pour coder un contour on peut évidemment choisir un pixel de départ sur le contour (dont on note les coordonnées image) puis on suit le contour par exemple dans le sens des aiguilles d'une montre et on note les différents codes de segments. Cette méthode est très sensible au bruit et erreurs sur le contour. Une solution peut consister à passer à une résolution inférieure et à coder sur cette nouvelle grille. On place donc une grille relativement « grossière » sur l'image et chaque point contour est placé sur le point de la grille le plus proche. On code alors les points de la grille obtenus en 4-connexité ou 8-connexité (figure III.10).

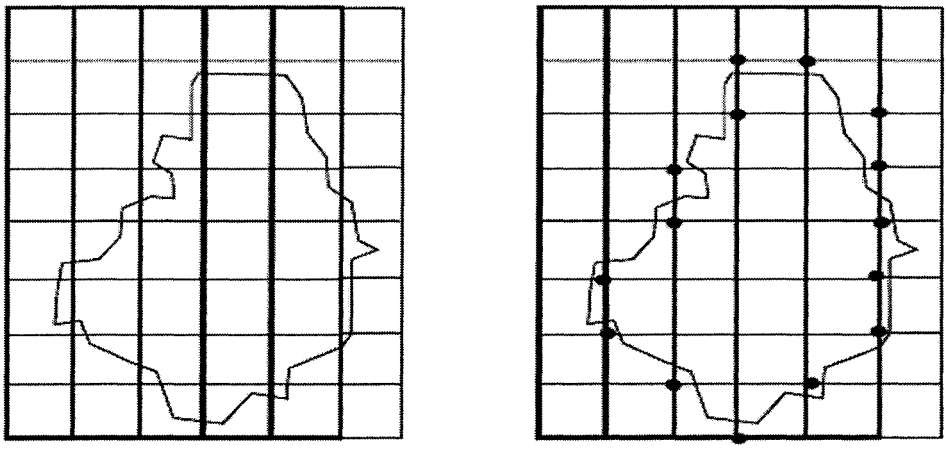


Figure III.10 : Codage des points de contour.

Les résultats suivants que l'on considère la 4-connexité ou la 8-connexité sont les suivants (figure III.11):

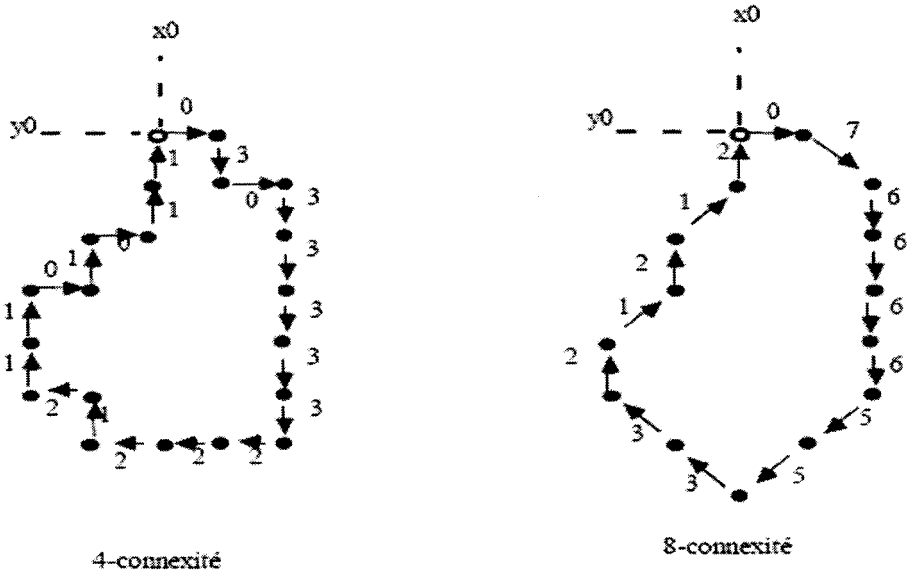


Figure III.11 : Résultats de codage des points de contour (4 et 8connexité).

soit les chaînes:

4-connexité: (x0,y0) , **03033333222121101011**

8-connexité: (x0,y0), **076666553321212**

On remarque bien sûr que la chaîne dépend du point de départ choisi. Si l'on veut normaliser le résultat on peut procéder par exemple ainsi : essayer tous les points de départ possibles et choisir la chaîne qui donne le « nombre entier » le plus petit, on évite ainsi le problème de mémoriser le point de départ.

6.Vectorisation d'une image

6.1 Introduction

La vectorisation est une phase très importante dans la segmentation d'images, elle consiste à approximer le squelette ou le contour d'une image par une suite de segments de droites (ou par des arcs de courbes). On y gagne en compacité et on se rapproche d'une description géométrique des objets. La topologie (connexité) des objets doit être conservée au cours de l'opération. Les techniques de vectorisation les plus connues sont :

- La transformée de Hough
- L'approximation polygonale
- La triangulation de Delanauy
- Etc...

6.2 Approximation polygonale

L'approximation polygonale consiste à transformer une chaîne de points connexes en une suite de segments de droites. En vision par ordinateur, l'approximation polygonale est une étape classique et même incontournable si les caractéristiques géométriques des objets doivent être prises en compte. Il va s'agir ici comme on a dit de remplacer un contour par une approximation polygonale, ce qui permettra de coder le contour par une suite de vecteurs.

Une possibilité pour réaliser cela est de considérer un quadrillage puis de noter tous les carrés contenant au moins un point contour. Le principe est alors de faire comme si on mettait un élastique dans cette suite de carrés et de voir quelle position il prend. Voir ci-dessous:

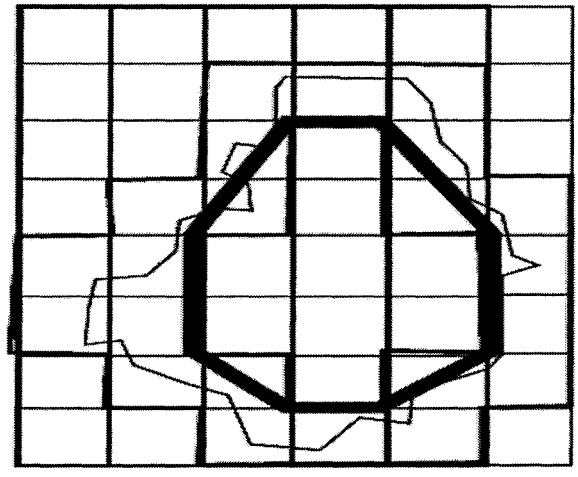


Figure III.12 : Approximation polygonale de contour.

Il est clair que plus le quadrillage est grossier plus l'approximation polygonale sera grossière. Cependant cette méthode n'est pas très simple à programmer.

Une autre approche consiste à partir d'un segment (par exemple le plus long possible joignant deux points du contour). Ce segment définit une droite qui découpe la région en 2. Dans

chaque demi-plan obtenu on cherche le segment perpendiculaire à la droite, partant de cette droite et joignant un point du contour le plus long possible.

Si cette longueur dépasse un seuil fixé à priori alors on choisit ce point contour comme nouvelle extrémité « polygonale ». On itère la méthode jusqu'à stabilité. L'exemple ci-dessous illustre cette méthode:

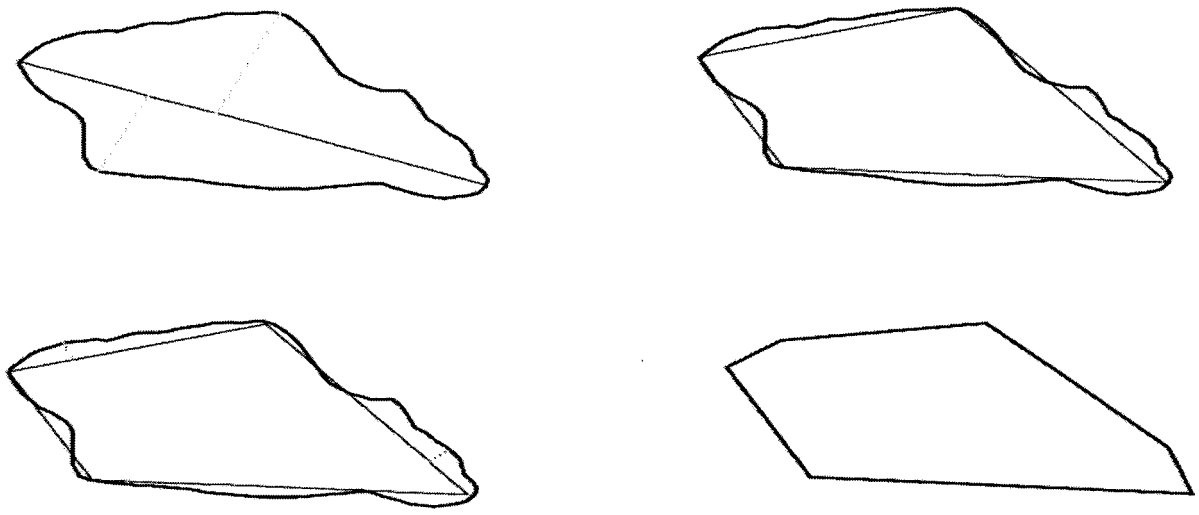


Figure III.13 : Les étapes d'une approximation polygonale.

Le résultat est l'ensemble des coordonnées des points extrémités.

Plus on fixe un seuil bas plus on sera prés du contour initial, mais aussi plus on aura de sommets donc de données à mémoriser. L'algorithme est le suivant :

- Trouver les 2 points les plus éloignés sur le contour chain-code. L'idée implantée pour trouver ce segment de droite, est de considérer la distance maximum du barycentre de l'objet au contour donnant un point de départ P0. On répète alors itérativement la séquence suivante, qui converge rapidement d'après les essais, vers le segment maximum [Pa Pb]:

1. Pi = P0
2. Pi+1 = P0
3. Chercher le point Pi+2 le plus éloigné de Pi+1 et appartenant au contour.
4. Pa = Pi+1; Pb = Pi+2

5. Si Pi = Pi+2 arrêt, sinon Pi = Pi+1; Pi+1 = Pi+2 et retour en 2

- Soit Pi = (xi,yi) les points ordonnés du polygone. Nous avons à la première itération, P1= Pa, P2 = Pb, P3 =Pa. Etant donné 2 points consécutifs du polygone, Pi et Pi+1, si la distance minimum de la droite [Pi Pi+1] à un point Pmax appartenant au contour, peut être supérieure à X, la valeur de tolérance de l'approximation, alors Pi+2 = Pi+1, Pi+1 = Pmax. On répète l'opération pour tous les Pi et Pi+1.

6.3 Transformée de Hough

C'est une technique "optimale" pour détecter les droites dans les images très bruitées. Cette technique ne dépend pas de la continuité des droites. Cependant, elle fournit des droites, pas des segments. Une équation de droite s'exprime comme :

$$y = m\,x + c \text{ ou } \qquad m = (y2-y1)/(x2-x1) \qquad c = y1 - m\,x1$$

Pour chaque point (x, y) de l'image, il y a un ensemble de valeurs possibles pour les paramètres m et c. Cet ensemble forme une droite d'équation c = -m x + y dans l'espace (m, c) appelé aussi espace de Hough.

Si des points de contrastes de l'image sont alignés, les droites correspondantes de l'espace de Hough passent toutes par un même point (m, c).

• *Calcul*

La transformée utilise un tableau h(m, c) de "cellules" comme illustrée dans la figure III.14. Les cellules de h(m, c) sont initialisées à 0. Pour chaque point (i, j) de l'image

si C(i, j) = 1, on repère la droite des couples (m, c) possibles

j = m i + c.

On ajoute 1 à chaque cellule de h(m, c) dans laquelle passe cette droite. Un maximum local de h(m, c) indique que des points de l'image sont alignés suivant la droite de paramètres correspondants (m, c).

• *Espace de Hough*

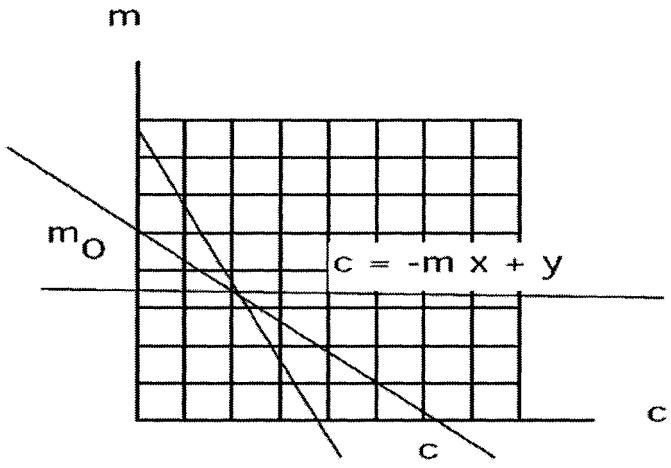


Figure III.14 : Espace de Hough.

• *Algorithme*

Pour chaque (i, j)

Si C(i, j) = 1:

Calculer les points d'interception avec les axes m, c:

$m_0 = -j/i$, $C_0 = j$

Pour tous les points (m, c) de la droite ((m_0, 0), (0, C_0))

h(m, c) := h(m, c) + 1

Les Maximum locaux de h(m, c) correspondent aux droites (m, c) de l'image.

Problème : les coefficients m, c ne sont pas uniformes en q.

Solution : On utilise les paramètres (r, q) de l'équation

$i \sin(\theta) - j \cos(\theta) + \rho = 0$.

• *Généralisation de Transformée de HOUGH*

Pour les cercles : une équation de cercle s'écrit :

$(x - a)^2 + (y - b)^2 = r^2$

On considère l'espace de Hough h(a, b, r).

Chaque point (x, y) de l'image correspond à un cône de l'espace (a, b, r).

Pour un rayon fixé, chaque point (x, y) correspond à un cercle de l'espace (a, b, r).

Idée de l'algorithme :

Pour chaque rayon r > 0 on trace les cercles de l'espace de Hough correspondant aux points de l'image. Lorsque tous les cercles se coupent en un même point, on a trouvé le bon rayon, les coordonnées (a, b) de ce point correspondent au centre du cercle.

6.4 Squeletisation

Cette approche consiste à « réduire » la région étudiée à un graphe représentant de façon simplifiée la forme de la région. Cette opération s'appelle une squelettisation de la région. La recherche d'un tel squelette de région peut être effectuée par la méthode de l'axe médian.

Soit une région R de contour C. Alors pour tout point p de la région on cherche le point du contour C le plus proche (au sens d'une distance particulière) et s'il y a plusieurs points de ce type alors le point p est dit appartenir au squelette de la région R.

Le résultat de la squelettisation dépend de la distance choisie (City-block, Euclidienne,...). Le temps de calcul est on s'en doute élevé car il faut calculer la distance de chaque point de la région à tous les points du contour c'est pourquoi on préférera des méthodes itératives consistant à amincir (éroder) petit à petit la région jusqu'au squelette. Exemple avec la distance Euclidienne (figure III.15):

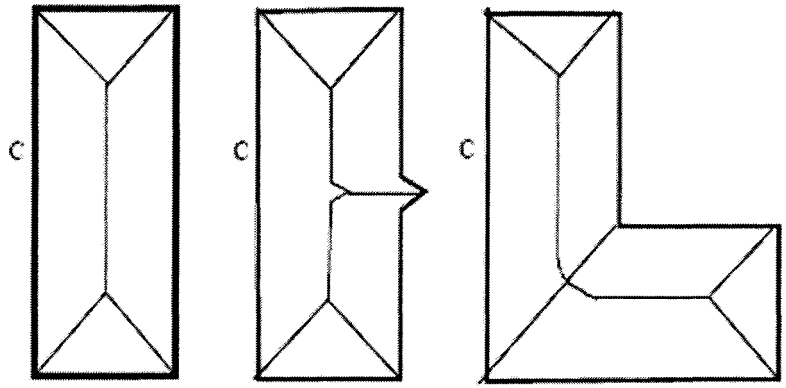


Figure III.15 : Amincissement de contour.

Nous allons maintenant présenter une méthode d'amincissement d'une région permettant d'obtenir un squelette de la région relativement rapidement:

Il s'agit d'une méthode fonctionnant par itération, chaque itération étant composée de deux passes (Tableau III.5).

Les points traités à chaque passe étant les points contour (c'est à dire les points de la région ayant au moins un de leurs 8-voisins ne faisant pas partie de la région).
Pour décrire la méthode nous allons supposer que la région a ses pixels qui valent 1 et les pixels extérieurs à la région valent 0 (le fonds). Alors, si l'on note les 8 voisins d'un point p1 tel qu'indiqué ci-dessous :

P9	**P2**	**P3**
P8	**P1**	**P4**
P7	**P6**	**P5**

Avec pi qui vaut 0 ou 1

Tableau III.5: Amincissement d'une région.

Chaque itération prend alors la forme suivante:

- Etape1 :

un point contour est effacé (mis à 0) s'il satisfait les conditions suivantes:

$2 <= N(p1) <= 6$, avec N(p1) nombre des voisins non nuls de p1.

$S(p1) = 1$

$p2 . p4 . p6 = 0$

$p4 . p6 . p8 = 0$

N(p1)= p2+p3+p4+p5+p6+p7+p8+p9

et S(p1) est le nombre de transitions 0-1 dans la séquence ordonnée p2, p3, p4, p5, p6, p7, p8, p9

Exemple :

```
0 0 1
1 p 0      N(p)=4 et S(p)=3
1 0 1
```

- Etape 2 :

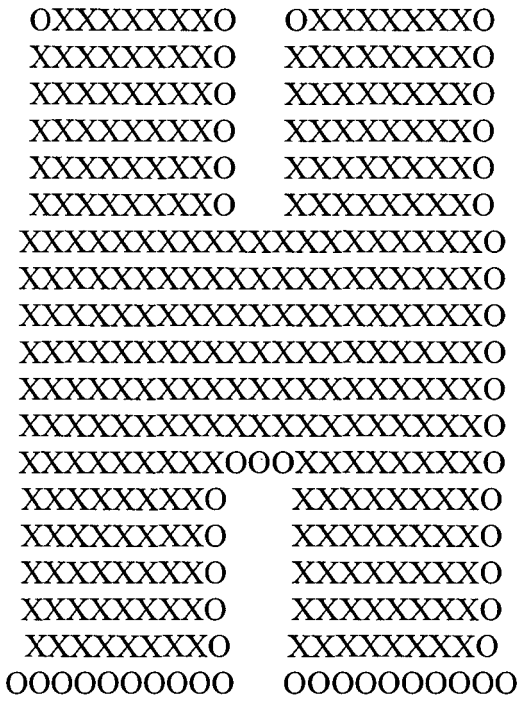
comme l'étape 1 mais les conditions N° 2 et 3 sont remplacées par

$p2 . p4 . p8 = 0$

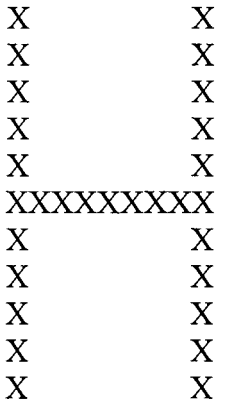
$p2 . p6 . p8 = 0$

Lors de l'itération les points de contours qui peuvent être effacés sont simplement notés et ils ne sont mis à 0 que lorsque tous les points du contour auront été passés en revue. Résultats sur l'image suivante:

PASSE 1 Itération1:



Résultat final:



7. Conclusion

Le traitement d'images est l'ensemble des techniques permettant de modifier une image dans le but de l'améliorer ou d'en extraire des informations. Ce traitement, souvent appelé prétraitement, il regroupe toutes les techniques visant à améliorer la qualité d'une image. De ce fait, la donnée de départ est l'image initiale et le résultat est également une image. La notion de qualité est une notion très subjective, assujettie à la réalisation d'un objectif. La qualité d'une image n'est pas forcément la même pour un ordinateur ou pour un opérateur humain. C'est la raison pour laquelle les techniques ne sont pas les mêmes. Par exemple, les techniques de segmentation et de vectorisation des images sont des techniques indispensables pour la compréhension et l'extraction de l'information que contient une image soit pour des raisons de compressions, d'indexation, de classification, de restauration, etc...Alors que toute analyse sur une image doit impérativement faire appel aux techniques du traitement d'images.

CHAPITRE
IV

Approche par les N-Grams : État de l'art

1. Introduction

Les modèles de langage N-Grams constituent particulièrement les modèles de langage de référence à l'analyse des corpus textuels et la reconnaissance de la parole et qui permet de modéliser des contraintes sur n caractères ou n mots à partir d'évènements observés sur un corpus d'apprentissage. Ces modèles donnent des résultats satisfaisants, car ils profitent d'une caractéristique commune à plusieurs langues.

Les N-Grams constituent un outil efficace pour le classement de textes. De nombreux travaux ont montré l'efficacité des N-Grams comme méthode de représentation des textes pour leur classification : recherche d'une partition en groupes homogènes, ou pour leur catégorisation : attribution d'un texte à une, ou plusieurs, catégorie(s) parmi une liste prédéterminée [DAM 95], [DUN 94], [MIL 99], [TEY 01], [CAV 94], [BIS 01].

2. Le codage en N-Grams de caractère

Bien qu'ayant été proposée depuis longtemps et utilisée principalement en reconnaissance de la parole, la notion de N-Grams de caractères prit davantage d'importance avec les travaux de [GRE 95] sur l'identification de la langue, de [DAM 95] sur le traitement de l'écrit. Ils prouvèrent que ce découpage, bien que différent d'un découpage en mots, ne faisait pas perdre d'information. Parmi les applications plus récentes des N-Grams on retrouve des travaux sur : l'indexation [MAY 98] ; l'hypertextualisation automatique multilingue avec les travaux de [HAL 98] qui, à travers une méthode de classification thématique de grandes collections de textes, indépendante du langage, construisent des interfaces de navigation hypertextuelle ; ou encore l'analyse exploratoire multidimensionnelle en vue d'une recherche d'information dans des corpus textuels [LEL 98].

Les N-Grams de caractères sont des séquences de *n* caractères consécutifs qui apparaissent dans le texte, le mot *« bon »* est un exemple de 3-grams (dite trigrams). Ces N-Grams constituent les dimensions d'un espace vectoriel dans lequel les documents sont représentés par des vecteurs. Les composantes de ces vecteurs sont les nombres d'occurrences des N-Grams qui apparaissent dans le texte. La liste exhaustive de ces N-Grams comprend toutes les séquences de *n* caractères produites par une fenêtre de *n* caractères de large déplacée le long du texte. Le mot *« bonjour »* produit les bigrams suivants : [bo], [on], [nj], [jo], [ou], [ur]. Le cas des trigrams donnes les séquences suivantes pour le même mot : [bon], [onj] , [njo], [jou], [our]. À cet effet, les N-Grams sont des bigrams pour n=2, des trigrams pour n=3, des quadrigrams pour n=4, etc....

Dans une approche avec découpage en N-Grams de caractères, contrairement aux approches avec découpage en mots, il n'est pas question d'utiliser la lemmatisation pour réduire le lexique. La lemmatisation (qui consiste à remplacer une forme fléchie par son lemme) est, d'une part, relativement lourde à mettre en oeuvre sur le plan informatique mais en plus, impose un traitement spécifique à chaque langue. Qui plus est, plusieurs lemmatiseurs ne semblent pas être en mesure de ramener des termes comme *informatisation*, *informatique*, et *informatiser* à un même concept qu'est l'informatique. Or souvent dans les corpus, on utilise des expressions ayant quasiment le même contenu informationnel comme, par exemple, dans les segments suivants : "l'informatisation de l'école", "informatiser l'école" et "introduire l'informatique à l'école". Le découpage des trois segments en N-Grams est suffisant pour classer les trois segments dans la même classe car, outre le mot école qui est redondant dans les trois expressions, les tri-grams inf, nfo, for, orm, rma, mat et ati, permettent par un calcul de similarité d'affirmer que c'est d'informatique dont il est question. Par ailleurs, les tri-grams susmentionnés apparaissent aussi dans le découpage des mots information, informationnel, etc., ce qui peut être considéré à juste titre comme du bruit, à moins bien sûr que l'on évoque une interprétation sémantique particulière de l'informatique comme étant une science de l'information. On comprend dès lors tout l'intérêt d'une plate-forme flexible qui nous permettrait de lemmatiser sans pour autant nous y obliger. Le cas présent en est la meilleure preuve.

2.1. L'intérêt du codage en N-Grams

• Comparativement à d'autres techniques, les N-Grams capturent automatiquement les racines des caractères et des mots les plus fréquents [GRE 95]. On n'a pas besoin de l'étape de recherche des racines lexicales (exemple : nourrir, nourri, nourrit, nourrissez, nourrissant, ... , nourriture, ... , nourrice, ...).

• Elles opèrent indépendamment des langues [DUN 94], contrairement aux systèmes basés sur les caractères et les mots dans lesquels il faut utiliser des dictionnaires spécifiques (féminin-masculin ; singulier-pluriel ; conjugaisons ; etc.) pour chaque langue. De plus, avec les N-Grams, on n'a pas besoin de segmentation préalable du texte en caractères ou mots ; ceci est intéressant pour le traitement des langues dans lesquelles les frontières entre caractères et mots ne sont pas fortement marquées, comme le chinois, ou encore pour les séquences ADN en génétique.

• Elles sont tolérantes aux fautes d'orthographes et aux déformations causées lors de l'utilisation des lecteurs optiques. Lorsqu'un document est scanné, la reconnaissance optique est souvent imparfaite. Par exemple, il est possible que le mot "*chapitre*" soit lu comme "*clapitre*". Un système basé sur les mots aura de mal à reconnaître qu'il s'agit du mot *chapitre*" puisque le mot est mal orthographié. Par contre, un système basé sur les N-Grams est capable de prendre en compte les autres N-Grams comme "*apit*", "*pitr*", etc. [MIL 99] montre que des systèmes de recherches documentaires basés sur les N-Grams ont gardé leurs performances malgré des taux de déformations de 30%, situation dans laquelle aucun système basé sur les mots ne peut fonctionner correctement.

• Enfin, ces techniques n'ont pas besoin d'éliminer les mots-outils (Stop Words) ni de procéder à la lemmatisation (Stemming). Ces traitements augmentent la performance des systèmes basés sur les mots. Par contre, pour les systèmes N-Grams, de nombreuses études [SAH 99] ont montré que la performance ne s'améliore pas après l'élimination des "Stop Words" et de "Stemming".

3. Quelques applications des N-Grams

3.1 Nettoyage de données

Le nettoyage des données a pour but de repérer et éliminer les erreurs et les états inconsistants d'une base afin d'augmenter la qualité de cette dernière. Ce problème émerge aussi bien dans des bases de données individuelles que lors de la fusion et l'intégration de plusieurs bases afin de constituer un entrepôt de données. Au-delà des erreurs simples comme les fautes de frappe dans les attributs enregistrés, des problèmes plus complexes doivent être résolus qui requièrent le balayage complet de la base. Un des soucis les plus étudiés s'avère l'élimination des tuples dupliqués et autres données redondantes tout en extrayant les informations utiles de ces tuples et en faisant disparaître les contradictions éventuelles.

Etant donné que la variété des problèmes rencontrés lors de l'intégration d'une ou plusieurs sources hétérogènes dans un entrepôt de données est très large [LEE 99], un des piliers des entrepôts est devenu la phase de nettoyage que l'on appelle aussi le processus de ETL (extraction, transformation, loading) Cette appellation montre bien le découpage des étapes nécessaires, notamment extraire les données des bases d'origine, les nettoyer, transformer et enfin procéder au chargement des données ainsi obtenues à l'entrepôt de données.

Comme dans un grand nombre des cas, les entrepôts de données sont exploités par des systèmes de support à la décision, la fiabilité des décisions dérivées dépendra très fortement de la qualité des bases initiales et de l'efficacité du processus ETL mis en œuvre.
Les N-Grams sont utilisés pour effectuer plusieurs taches telles que :

- *Détection des fautes de frappe.*
- *Elimination des redondances (minimalité et complétude).*

3.2 Catégorisation de documents multilingues

Elle a pour but de déterminer la langue dans laquelle est écrit un document, et le sujet qu'il traite, sans besoin d'information a priori sur le document. Cette méthode permet de catégoriser et de retrouver les documents d'une grande collection de documents multilingues (méthode de L'Acquaintance [HUF 94]). Cette méthode est la suivante : après avoir normalisé la casse des documents, on extrait tous les N-Grams qu'ils contiennent. Le nombre de N-Grams trouvés est le

nombre de dimension de l'espace. On affecte un vecteur de cet espace à chaque document, dont les coordonnées sont les nombres d'occurrences des N-Grams.

Pour mesurer la similarité des documents, [DAM 95] fait l'hypothèse que deux documents dont les vecteurs de N-Grams sont similaires est susceptible de traiter des sujets proches, et les documents dont les vecteurs sont dissimilaires sont susceptibles d'avoir des contenus éloignés. La similarité est évaluée en mesurant le cosinus de l'angle formé entre les vecteurs. Il est dès lors possible de catégoriser les documents selon leur langue, de déterminer, parmi ceux qui sont écrits dans la même langue, ceux qui traitent de sujets similaires. Quand un document existant est utilisé comme une requête, il est donc possible de retrouver des documents similaires.

L'approche de l'Acquaintance est intéressante pour le modèle de document et le modèle de requête que la méthode définit. La fonction de correspondance, elle, est classique puisqu'elle consiste en un cosinus de l'angle formé par deux vecteurs.

3.3 Reconnaissance automatique de la parole

Les modèles de langage N-Grams, qui constituent les modèles de langage de référence en reconnaissance de la parole, modélisent des contraintes sur n mots à partir d'évènements observés sur un corpus d'apprentissage. Ces modèles donnent des résultats satisfaisants car ils profitent d'une caractéristique commune à plusieurs langues, dont le français, qui exercent des contraintes locales fortes sur l'ordre des mots. Ils arrivent ainsi à résumer simultanément une grande partie des connaissances syntaxiques et sémantiques issues de l'observation du corpus d'apprentissage.

Malheureusement, l'utilisation de ces modèles probabilistes est confrontée à plusieurs difficultés (manque d'informations statistiques, portée des contraintes modélisées trop courtes pour certains phénomènes linguistiques).

Afin de pallier les difficultés des modèles N-Grams en mots, un modèle hybride été proposé et qui combine un modèle de langage n-gram avec des grammaires régulières locales: les connaissances linguistiques apportées par ces grammaires sont directement intégrées dans le modèle.

3.4 Désambiguïsations lexicales

La désambiguïsation lexicale s'effectue toujours en utilisant l'information du contexte du mot à désambiguïser. Cette information peut être enrichie par un certain nombre d'annotations (étiquette morphosyntaxique, lemmatisation, etc.). Cette information peut également être utilisée conjointement avec des bases de connaissances externes. Dans tous les cas, il n'est pas possible d'utiliser toute l'information disponible car elle est bien trop bruitée, par exemple le mot *barrage* peut référer à un barrage hydraulique, un barrage de police, un barrage de guitare, etc.. Il faut donc se focaliser sur un certain nombre d'indices. Le choix de ces indices, déterminé par ce que nous appelons des critères de désambiguïsation lexicale, est primordial et constitue un enjeu important en désambiguïsation lexicale. Les N-Grams ont pour objectif de réaliser une étude systématique et approfondie de critères pour la désambiguïsation lexicale automatique et permettre d'étudier des critères basés sur des cooccurrences de mots, et plus généralement de n-grammes (juxtaposition de un ou plusieurs mots), en tentant de répondre aux multiples questions que l'utilisation de tels critères soulève, comme la taille et la symétrie des contextes à considérer, l'importance de la lemmatisation, de l'ordre des mots, des mots grammaticaux, de la taille des N-Grams utilisés, etc.

La désambiguïsation lexicale est très utile voire indispensable dans un grand nombre de domaines de recherche en traitement automatique des langues naturelles, on cite par exemple :

• *Restauration de l'accentuation*

[YAR 94a], [YAR 94b] développe des algorithmes qui permettent de restaurer les accents sur des textes ayant perdu toute accentuation, de corriger des fautes d'accentuation dans le cadre des logiciels de correction orthographique et grammaticale et de s'affranchir de la saisie des accents automatiquement ajoutés lors de la frappe du texte (cette technique permet notamment l'utilisation des claviers américains dépourvus de touches d'accentuation). Cette tâche est, à plus d'un titre, un bon exercice pour un algorithme de désambiguïsation lexicale :

– Le problème est représentatif des difficultés rencontrées dans la résolution des ambiguïtés lexicales.

– Ce type de problème permet de s'affranchir de la difficulté de trouver des textes d'apprentissage étiquetés puisqu'il suffit d'utiliser des textes correctement accentués (très largement disponibles) et de supprimer les accentuations pour générer des corpus de test.
– Ce problème débouche directement sur plusieurs types d'applications pratiques, voire commerciales.

• *Recherche d'informations*

Lever l'ambiguïté des mots d'une requête peut permettre d'affiner la recherche. Par exemple, si nous cherchons des textes traitant des rayons laser, il faut ignorer les textes traitant des rayons de soleil ou encore des rayons de bicyclette.

Les travaux récents en désambiguïsation lexicale s'inspirent parfois dans plusieurs travaux. En effet, définir si un sens particulier s'applique à l'instance d'un mot est, dans une certaine mesure, analogue à savoir si un document donné est une réponse pertinente à la requête formulée.

Actuellement, les travaux en recherche d'informations utilisent rarement les techniques d'étiquetage morphosyntaxique ou d'étiquetage lexical. La raison en est que ces techniques ne sont pas assez rapides, robustes ou portables, et qu'elles n'apportent pas toujours d'amélioration substantielle. D'ailleurs, plusieurs expériences ont montré que l'ambiguïté du sens des mots ne dégrade pas beaucoup les performances des algorithmes [KRO 92], [SAN 94]. De plus, il se produit une désambiguïsation implicite lorsque plusieurs mots clefs d'une requête concordent avec plusieurs mots dans un document [RES 97]. En fait, [SAN 94] montre que lorsque la désambiguïsation lexicale est très précise, elle apporte un plus en recherche d'informations, sinon elle dégrade les performances.

D'une certaine manière, les recherches en recherche d'informations ont, pour l'instant, plus apportées à celles en désambiguïsation lexicale que le contraire. En effet, les recherches ont progressé en utilisant des méthodes statistiques sur des documents dont la structure linguistique est ignorée. Or, c'est vers ce type d'approches que les recherches en désambiguïsation lexicale se tournent actuellement.

3.5 Hypertextualisation automatique multilingue

Une méthode de classification thématique de grandes collections de textes été présentée qui est indépendante du langage, permettant de créer des interfaces de navigation hypertextuelles dans ces collections, quelle que soit la langue utilisée. Cette méthode caractérise les textes par leurs fréquences de N-Grams (séquences de n-catactères consécutifs).

Plusieurs variantes de construction des vecteurs-textes, et de pondération de ceux-ci, sont présentées et comparées aux résultats obtenus avec une représentation des documents par la fréquence de leurs termes d'indexation. L'analyse de ces données est assurée par un modèle neuronal.

3.6 Analyse de grands corpus textuels

La qualité d'une analyse de données textuelle repose principalement sur la qualité de la phase d'extraction de termes qui précède (indexation, au sens général). Les techniques de détection de termes composés produisent des listes immenses, dont l'accès est indispensable pour exercer un contrôle éditorial sur le vocabulaire d'indexation [HAL 98] a présenté un outil de détection d'"inclusion floue" d'une chaîne de caractères dans une chaîne plus grande, utile dans ce contexte, et basé sur un indice d'inclusion calculé à partir des fréquences de N-Grams, puis sur un indice de séquences.

4. Conclusion

L'indexation de l'information, le traitement des langues naturelles et le traitement de la parole deviennent une contrainte dont il faut tenir compte dans le développement de chaînes de traitement dédiées à l'analyse et au traitement de l'information textuelle et sonore. Si les classifications numériques ont prouvé leur efficacité dans le traitement de gros corpus, aucune réflexion sérieuse n'a été entamée quant à leur capacité à traiter des corpus peu importe leur nature. À cet effet, Les N-Grams constituent un outil efficace pour le classement de textes et de la parole. De nombreux travaux cités dans ce chapitre ont montré l'efficacité des N-Grams comme méthode de représentation, particulièrement de corpus textuels, pour leur classification. L'avantage de cette méthode est qu'elle est indépendante de la nature de l'information.

CHAPITRE
V

Résultats et interprétations

1. Introduction

SYCLIM (Système de Classification d'images) est le prototype que nous avons développé pour la classification d'images. Il prend en entrée des images numériques extraites d'une base de données d'images représentant des animaux, des objets ou encore des personnes, etc. Il produit en sortie des classes de similarités qui contiennent les images semblables. La classification portera essentiellement sur la similarité des images étant donnée la régularité des distributions des valeurs d'intensité des couleurs dans les pixels des images. Le classifieur proposé est un réseau de neurone de type ART dont nous avons expliqué le principe au chapitre III de ce mémoire.

2. Architecture du prototype SYCLIM

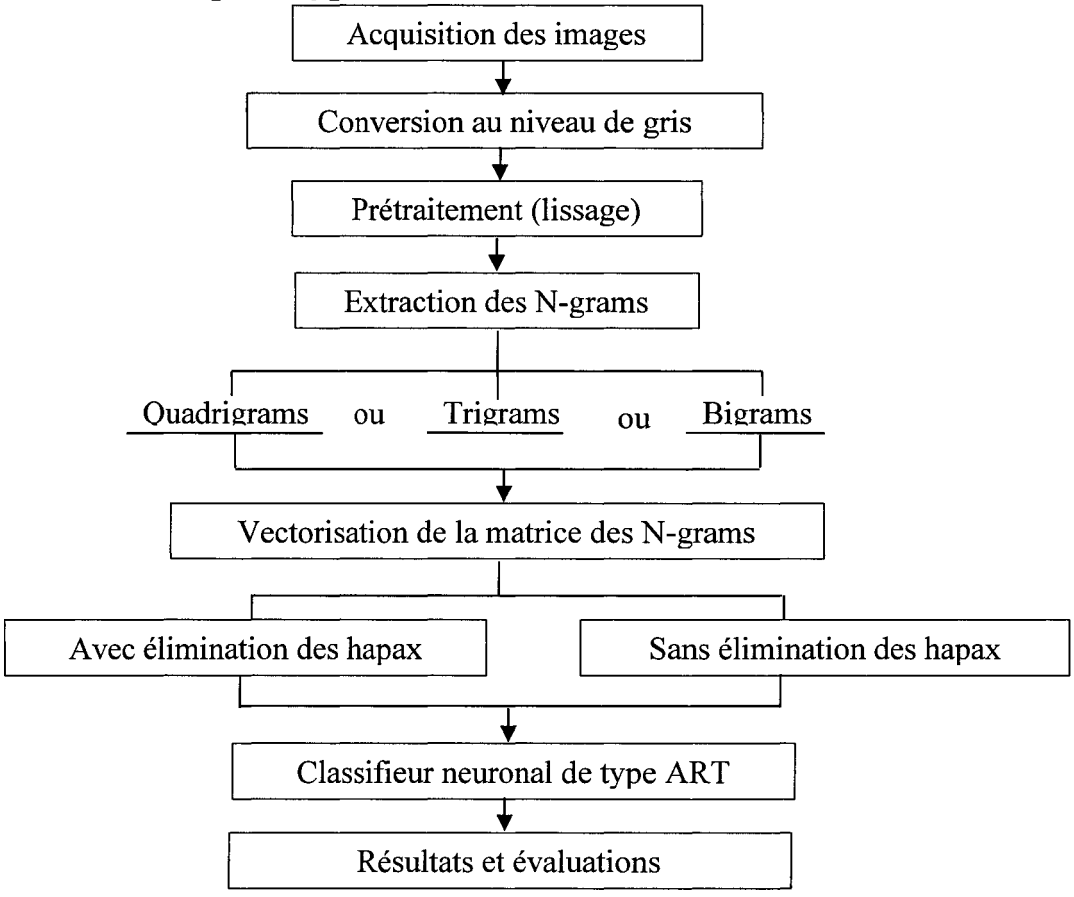


Figure V.1 : Architecture du prototype SYCLIM.

3. Description des différentes taches de SYCLIM

3.1 Conversion des images couleurs en niveau de gris

Pour les images couleurs, un pixel dispose généralement des trois composantes RGB (Red, Green, Blue). Un pixel gris a ses trois valeurs RGB identiques. Une méthode simple pour convertir une image couleur en niveaux de gris pourrait être : calculer la moyenne des trois composantes RGB et utiliser cette valeur moyenne pour chacune des composantes.

$$Gris = (Red + Green + Blue) / 3$$

La C.I.E (Commission Internationale de l'Éclairage) propose, de caractériser l'information de luminance (la valeur de gris) d'un pixel par la formule :

$$Gris = 0.299*Red + 0.587*Green + 0.114*Blue$$

Son importance en est pas moins considérable puisque l'espace de niveaux de gris d'une image est une étape essentielle de tout traitement car la complexité du programme et le temps de calcul s'en trouvent réduits.

Voici un exemple qui démontre le passage d'une image couleur RGB en une image au niveau de gris :

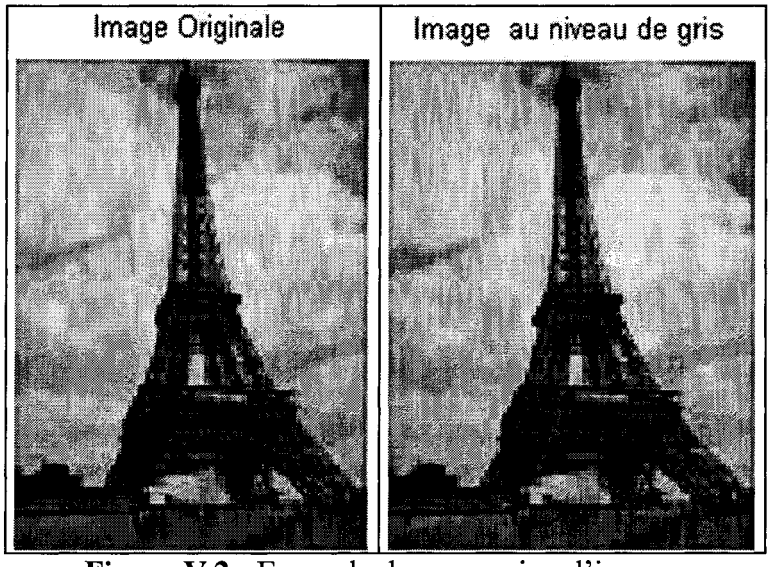
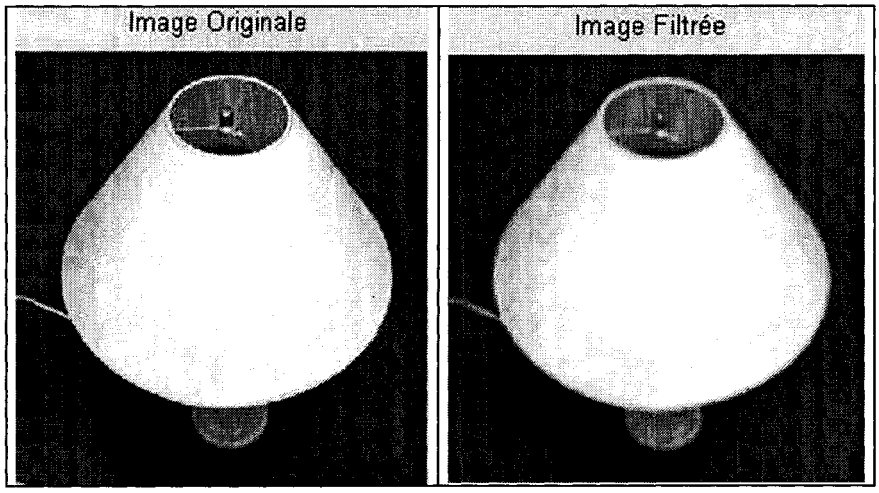


Figure V.2 : Exemple de conversion d'image.

3.2 Lissage des images

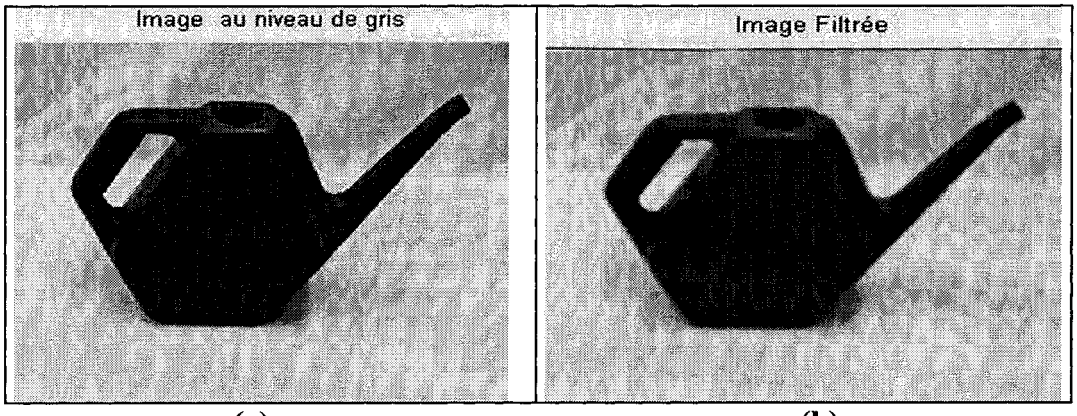
Le filtre appliqué aux les images de notre base de données est de type passe-bas, il consiste à atténuer les composantes de l'image ayant une fréquence haute (pixels foncés). Ce type de filtrage est généralement utilisé pour atténuer le bruit de l'image, c'est la raison pour laquelle on parle habituellement de lissage. Les filtres par moyennage sont de type passe-bas dont le principe est de faire la moyenne des valeurs des pixels avoisinants. Le résultat de ce filtre est une image plus floue. Le principe de cette opération est expliqué au chapitre 03 de ce mémoire. L'exemple ci-dessous montre l'effet d'opération de filtrage par moyennage pour une image couleur :



(a) **(b)**

Figure V.3 : Exemple de lissage d'image couleur, (a) Image originale (b) Image lissée (filtrage par la moyenne).

Exemple d'une image d'un *Sceau.* L'image est aux niveaux de gris :



(a) **(b)**

Figure V.4 : Exemple de lissage d'images aux niveaux de gris, (a) Image originale (b) Image lissée (filtrage par la moyenne).

3.3 Extraction des N-grams

Comme c'est définit pour les N-Grams de caractères dans le cas d'une analyse textuelle, les N-grams dans le cas des images représentent une suite de n valeurs d'intensités de pixels que nous noterons dorénavant NGVIP. Les BGVIP (Bigrams des valeurs d'intensités des pixels) représentent les N-grams de valeurs de 2 pour n, les TGVIP (Trigrams des valeurs d'intensités des pixels) pour n=3, etc.

La figure ci-dessous représente les séquences des BGVIP et TGVIP pour une image.

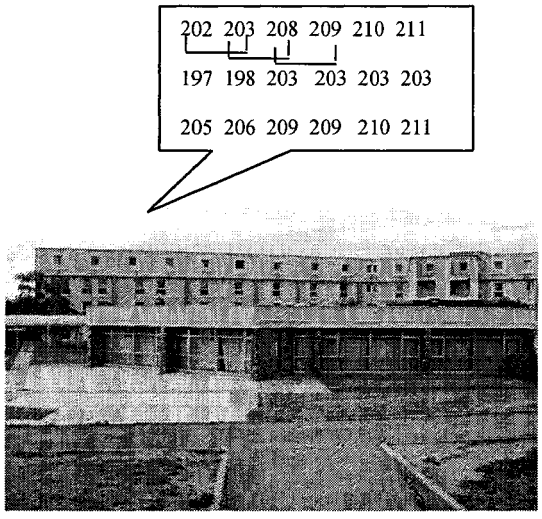


Figure V.5 : Exemple d'extraction des NGVIP.

Bigrams : 202203, 203208, 208209, 20921, etc.

Trigrams : 202203208, 203208209, 208209210, etc.

Quadrigrams : 202203208209, 203208209210, etc.

3.5 Vectorisation de la matrice des N-grams

Pour toutes les images de notre base de données, les N-Grams NGVIP des valeurs d'intensité des pixels sont extraites. Après, les images sont représentées par une liste des NGVIP. La totalité des images sont alors structurées dans une matrice n x m (n = le nombre d'images; m = le nombre de NGVIP contenu dans la totalité des images). On indique également la fréquence d'un NGVIP dans une image. Cette matrice sera employée comme entrée au classificateur ART. La représentation vectorielle de cette matrice est montrée dans le tableau (V.1) :

	ngram 1	*ngram 2*	*ngram j ...*	*ngram m*
Image 1	Freq(1,1)	Freq(1,2)	Freq(1,j) ...	Freq(1,m)
Image 2	Freq(2,1)	Freq(2,2)	Freq(2,j) ...	Freq(2,m)
Image 3	Freq(3,1)	Freq(3,2)	Freq(3,j) ...	Freq(3,m)
Image i	Freq(i,1)	Freq(i,2)	Freq(i,j) ...	Freq(i,m)
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
Image n	Freq(n,1)	Freq(n,2)	Freq(n,j) ...	Freq(n,m)

TableauV.1 : Représentation vectorielles des images.

3.6 Élimination des hapax

Il est dans la tradition de la classification numérique d'enlever les unités d'information dont la fréquence est égale à 1(hapax) pour deux raisons fondamentales:

- Ces unités ne sont pas considérées comme porteuses d'information.
- Puisque ces unités sont souvent nombreuses. Leur suppression réduit le temps nécessaire au calcul nécessité par le processus de classification et permet une économie en terme d'occupation des ressources mémoire ou de temps d'exécution.

Pour l'évaluation de nos résultats, nous traitons les deux cas :

- Cas 1 : Maintenir les hapax.
- Cas 2 : Éliminer les hapax.

3.7 Le classifieur ART

Les images représentées dans la matrice obtenue à l'étape précédente sont comparées entre elles au moyen d'un classificateur numérique. Le choix du classificateur devient un paramètre dans la conception de la chaîne de traitements. Le paramètre étant défini, commence alors l'exécution du module associé à la classification. Les images qui sont semblables, étant donné une certaine fonction de similitude (dépendamment du classificateur choisi), seront regroupées

dans les mêmes classes. Tout en simplifiant, on peut indiquer que deux images sont reconnues semblable si elles sont constituées des mêmes NGVIP avec des fréquences presque identiques.

Pour l'évaluation qui suivra nous avons choisi le réseau neuronal ART, son algorithme et son architecture sont expliqués en détail dans le chapitre II.

D'un point de vue concret, ce réseau emploie un paramètre appelé *Rho* (indice de similarité) dont la valeur est comprise entre 0 et 1. Plus sa valeur tend vers 0, moins il est discriminant. À l'opposé quand sa valeur tend vers 1, il devient plus discriminant et par conséquent, produit un grand nombre de classes de similitudes. Pour nos évaluations *Rho* prend la valeur de 0,2.

4. Interprétation des résultats

Notre approche est semi-automatique. Le dernier mot revient naturellement à l'utilisateur. Comparer notre approche à d'autres approches automatiques serait maladroit et nous devons par conséquent considérer notre évaluation sous un point de vue différent.

Avec cette intention, nous avons pris dans une première série d'évaluation une base de données de 23 images (nos images sont de format JPEG et d'une résolution 320*200, voir figure V.6). Certaines de ces images représentent le même objet vu sous différents angles. D'autres images représentent des objets de la nature comparable (non nécessairement les mêmes images) qu'il est possible de classer ensemble. Un premier résultat fondamental est noté:

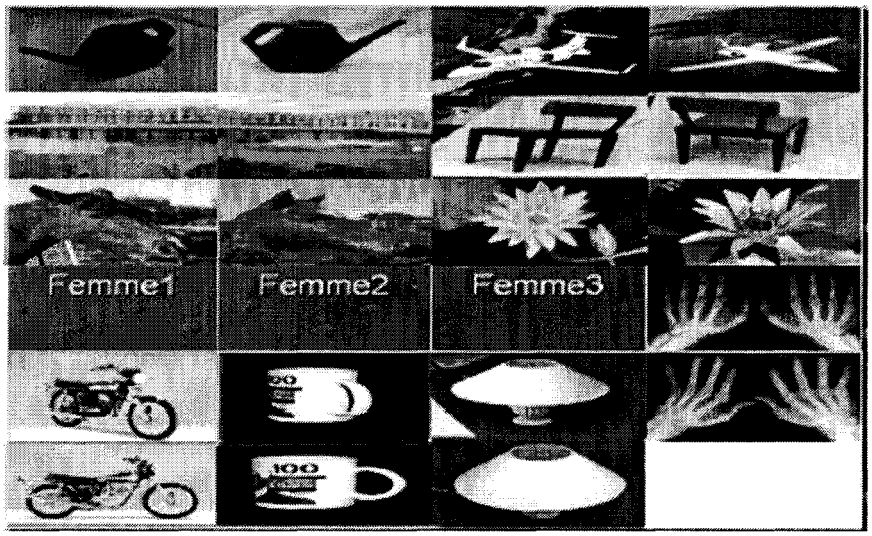


Figure V.6 : La base de données avec 23 images.

La vectorisation des images, en considérant un découpage en N-grams de valeurs 'intensités des pixels, permet d'obtenir une classification d'une très grande qualité que ce soit d'un point de vue qualitatif ou quantitatif. Pour obtenir cette observation, plusieurs évaluations sur la même base de données des images ont été effectuées.

Dans une première évaluation nous avons employé les bigrams de valeurs d'intensités des pixels (BGVIP) et seulement les niveaux de gris ont été pris en compte. Deux variantes de cette première évaluation ont été étudiées. Dans la première, nous avons considéré la suppression du BGVIP avec une fréquence égale à 1 (hapax). Dans la seconde nous les avons conservés. Nous avons obtenu les résultats suivants:

Dans le premier cas, 11 classes ont été obtenues. Trois de ces classes contiennent chacune un objet qui devrait avoir été regroupé avec un autre objet. Ainsi, les classes 2 et 3, 4 et 5, 8 et 9 contiennent des objets semblables. Deux classes contiennent : deux sous-classes pour la première ; et trois sous-classes pour la seconde. Ces deux classes sont légèrement bruitées aussi.

Dans le deuxième cas, 13 classes ont été obtenues. Cinq de ces classes étaient parfaites, les images qu'elles contenaient étaient parfaitement semblables. Quatre autres classes contenaient chacune un objet qui aurait dû être classé ailleurs. Ainsi les classes 1 et 2 d'une part et 11 et 12 d'autre part contiennent les objets semblables. Enfin les classes 6 et 8 contiennent chacune un objet qui n'a aucun rapport avec les autres objets de ces classes.

Ce que nous notons ici est que la suppression du hapax (BGVIP avec une fréquence égale à 1) a un effet pervers sur le résultat de la classification. Contrairement à la classification textuelle où les hapax sont considérés comme non-porteurs d'une capacité discriminante significative, dans le cas de la classification des images les hapax sont donc très significatifs.

Nous pouvons récapituler ces premiers résultats d'évaluation dans le tableau V.2 (les images sont au niveau du gris):

	Classes		
	parfaites	*Légèrement bruitées*	*bruitées*
hapax eliminés (11 classes)	0	2	0
hapax maintenus (13 classes)	5	0	2

Tableau V.2 : Résultats d'évaluation (images en niveaux de gris, 23 images).

Le tableau V.3 illustre les résultats d'évaluation de notre première base de données (les 23 images en niveaux de gris) avec les classes correspondantes aux images pour les cas des hapax éliminés et maintenus. On remarque que les deux classes bruitées la ou les hapax sont éliminés sont la classe du bâtiment et celle de la chaise. Pour le cas ou les hapax sont maintenus, cinq classes sont parfaites, se sont pour les images d'avions, bâtiments, chaises, mains et motos.

Arrosoir 1	1
Arrosoir 2	1
Avion 1	1
Avion 2	1
Bâtiment 1	2
Bâtiment 2	3
Chaise 1	4
Chaise 2	5
Cheval 1	11
Cheval 2	6
Femme1	6
Femme2	6
Femme3	6
Fleur1	1
Fleur2	7
Lampe1	6
Lampe1	6
Main1	8
Main2	9
Moto1	10
Moto2	6
Tasse1	6
Tasse2	6

(a)

Arrosoir 1	1
Arrosoir 2	2
Avion 1	3
Avion 2	3
Bâtiment 1	4
Bâtiment 2	4
Chaise 1	5
Chaise 2	5
Cheval 1	6
Cheval 2	6
Femme1	6
Femme2	13
Femme3	6
Fleur1	7
Fleur2	8
Lampe1	8
Lampe1	8
Main1	9
Main2	9
Moto1	10
Moto2	10
Tasse1	11
Tasse2	12

(b)

TableauV.3 : Résultats d'évaluation avec les classes correspondantes aux images (a) hapax éliminés, (b) hapax maintenus.

Dans notre deuxième évaluation, nous avons maintenu le découpage des images en bi-grams BGVIP. Cependant nous avons considéré les images en couleur. 13 classes ont été obtenues, neuf de ces classes étaient parfaites. Les quatre autres classes contenaient chacune un objet. Ces objets dans une situation parfaite auraient dû former deux classes de similitude. Cependant, le plus intéressant avec cette évaluation est qu'aucune classe bruitée n'a été obtenue.

Il est possible de dire à travers ces résultats, que le fait de considérer les images en couleurs dans trois dimensions soit beaucoup plus discriminant que de les considérer aux niveaux de gris.

Dans notre troisième évaluation, nous avons découpé nos images en trigrams de valeurs de l'intensité des Pixel (TGVIP). 16 classes ont été obtenues, parmi lesquelles quatre étaient parfaites. Seulement une seule classe était bruitée. Toutes les autres classes contenaient un seul objet. Ce résultat n'est pas inintéressant, il nous permet d'apprécier le fait que l'utilisation du TGVIP n'augmente le nombre de bruit. Cependant, il faut souligner qu'une telle taille de NGVIP devient plus contraignante quant à savoir si deux images sont semblables.

Ainsi, si un découpage en BGVIP donnerait deux images i1 et i2 dans la même première classe et i3 et i4 dans la même deuxième classe, tandis qu'un découpage en TGVIP classerait seulement les images i1 et i2 dans la même classe, on pourrait supposer que les images i1 et i2 sont beaucoup plus semblables que les images i3 et i4.

Nous pouvons récapituler ces deuxième et troisième résultats d'évaluation dans le tableau V.3 (les images sont en couleur):

	Classes		
	parfaite	*légèrement bruitées*	*Bruitées*
Découpage en BGVIP (13 classes)	9	0	0
Découpage en TGVIP (16 classes)	4	0	1

TableauV.4 : Résultats d'évaluation (images couleur, 23 images).

Le tableau V.5 illustre les résultats d'évaluation de notre première base de données (les 23 images sont en couleur) avec les classes correspondantes aux images pour les cas de découpage en BGVIP et en TGVIP. Dans le cas du découpage en BGVIP, les classes représentées par les images des bâtiments, chaises, chevaux, femmes, fleurs, lampes, mains, motos et tasses sont parfaites. Alors que dans le cas du découpage en TGVIP, les classes parfaites sont celles des images des femmes, lampes, motos et les tasses.

Arrosoir 1	1
Arrosoir 2	2
Avion 1	3
Avion 2	4
Bâtiment 1	5
Bâtiment 2	5
Chaise 1	6
Chaise 2	6
Cheval 1	7
Cheval 2	7
Femme1	8
Femme2	8
Femme3	8
Fleur1	9
Fleur2	9
Lampe1	10
Lampe1	10
Main1	11
Main2	11
Moto1	12
Moto2	12
Tasse1	13
Tasse2	13

(a)

Arrosoir 1	1
Arrosoir 2	2
Avion 1	3
Avion 2	4
Bâtiment 1	5
Bâtiment 2	6
Chaise 1	7
Chaise 2	8
Cheval 1	8
Cheval 2	8
Femme1	9
Femme2	9
Femme3	9
Fleur1	16
Fleur2	10
Lampe1	11
Lampe1	11
Main1	12
Main2	13
Moto1	14
Moto2	14
Tasse1	15
Tasse2	15

(b)

TableauV.5 : Résultats d'évaluation avec les classes correspondantes aux images **(a)** Découpage en BGVIP, **(b)** Découpage en TGVIP.

Une deuxième série d'évaluations porte sur une base de données plus élargie (figure V.7). La base de données contient 52 images (les images sont toujours au format JPEG et d'une résolution 320*200). Deux variantes ont été considérées. La première avec le découpage des

images en bigrams *BGVIP*, tandis que la seconde avec un découpage en trigrams *TGVIP*. Les hapax sont maintenus dans les deux cas. L'indice de similarité choisi est égal à 0.2 (valeur de *Rho*). Les images sont en couleur.

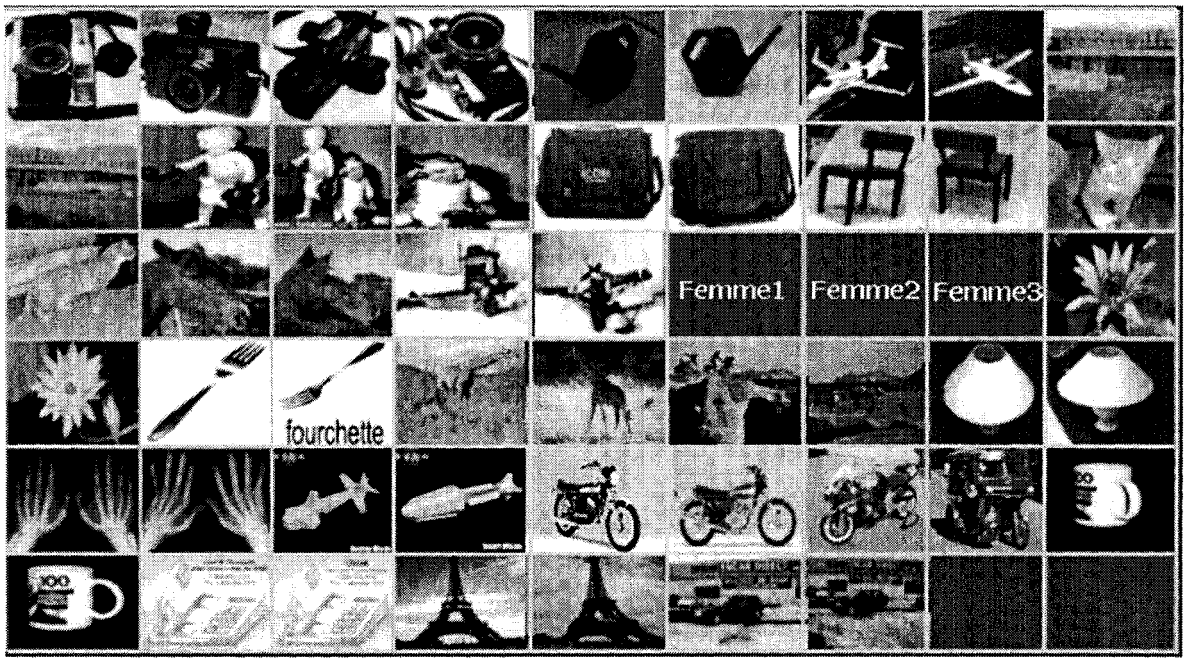


Figure V.7 : La base de données avec 52 images

Dans le premier cas, nous avons obtenu 23 classes. Sept classes étaient absolument parfaites. Chacune de ces classes recueille les objets que nous avons considérés comme parfaitement semblables et aucun bruit dans ces classes. Huit classes obtenues sont légèrement bruitées. Certaines contiennent un élément qui représente un "bruit", d'autres ne contiennent pas un élément qui aurait être dans cette classe. Cependant, ces 8 classes restent suffisamment cohérentes pour être utilisables. 5 classes contiennent chacune seulement une image. En conclusion, trois classes sont trop bruitées et peuvent être considérées comme étant inutilisables.

Dans le deuxième cas, nous avons obtenu 27 classes, parmi lesquelles 6 classes sont légèrement bruitées et deux autres classes trop bruitées. Les autres classes contiennent chacune seulement une image. Ce que nous pouvons dire est que la division d'image dans les trigrams *TGVIP* augmente le taux de discrimination.

Nous pouvons récapituler nos résultats de cette deuxième série d'évaluations dans le tableau V.4:

	Classes		
	parfaites	*Légèrement bruitées*	*bruitées*
Découpage en BGVIP (23 classes)	7	8	3
Découpage en TGVIP (27 classes)	0	6	2

Tableau V.6: Résultats d'évaluation (52 images couleurs).

Le tableau V.6 illustre les résultats d'évaluation de notre première base de données (les 52 images sont en couleur) avec les classes correspondantes aux images pour les cas de découpage en BGVIP et en TGVIP. Dans le découpage en BGVIP :

- Les images arrosoirs, bâtiments, chaises, chevaux, îles, missiles, et tours sont parfaitement classées,
- Les classes légèrement bruitées sont pour celles des voitures, motos, mains, fleurs, hélicoptères, appareils, bébés et les femmes,
- Les classes bruitées sont pour les images représentant les chats, girafes et cartables.

	Rho=0.2		Rho=0.2
Appareil1	1	Fleur1	13
Appareil2	1	Fleur2	13
Appareil3	2	Fourchette1	12
Appareil4	1	Fourchette2	17
Arrosoir1	3	Girafe1	11
Arrosoir2	3	Girafe2	13
Avion1	4	Ile1	14
Avion2	5	Ile2	14
Batiment1	6	Lampe1	19
Batiment2	6	Lampe2	17
Bébé1	7	Main1	4
Bébé2	5	Main2	4
Bébé3	7	Missile1	16
Cartable1	8	Missile2	16
Cartable2	2	Moto1	17
Chaise1	9	Moto2	17
Chaise2	9	Suzuki1	18
Chat1	11	Suzuki2	23
Chat2	13	Tasse1	4
Cheval1	10	Tasse2	20
Cheval2	10	Téléphone1	21
hélicoptère1	12	Téléphone2	17
hélicoptère2	12	Tour1	22
Femme1	15	Tour2	22
Femme2	15	Voiture1	23
Femme3	1	Voiture2	23

(a)

	Rho=0.2		Rho=0.2
Appareil1	1	Fleur1	1
Appareil2	1	Fleur2	13
Appareil3	1	Fourchette1	1
Appareil4	1	Fourchette2	1
Arrosoir1	1	Girafe1	1
Arrosoir2	1	Girafe2	5
Avion1	1	Ile1	14
Avion2	2	Ile2	15
Batiment1	3	Lampe1	16
Batiment2	24	Lampe2	27
Bébé1	11	Main1	1
Bébé2	6	Main2	1
Bébé3	7	Missile1	18
Cartable1	4	Missile2	19
Cartable2	1	Moto1	1
Chaise1	4	Moto2	10
Chaise2	5	Suzuki1	1
Chat1	8	Suzuki2	1
Chat2	9	Tasse1	1
Cheval1	1	Tasse2	20
Cheval2	25	Téléphone1	21
hélicoptère1	1	Téléphone2	22
hélicoptère2	26	Tour1	23
Femme1	12	Tour2	17
Femme2	1	Voiture1	1
Femme3	1	Voiture2	5

(b)

TableauV.7 : Résultats d'évaluation avec les classes correspondantes aux images (52 images couleur) : **(a)** Découpage en BGVIP, **(b)** Découpage en TGVIP.

Dans le cas du découpage en TGVIP:

- Les classes légèrement bruitées: appareils, arrosoirs, femmes, fourchettes, mains et les susukis.
- Les classes bruitées sont : girafes et voitures.

L'interprétation de nos résultats nous laisse penser que le lissage et le maintien des hapax ont un effet positif sur la qualités de nos résultats obtenus puisque le lissage est généralement a pour but de réduire les parasites ou interférences provenant soit à qualité du capteur, qualité du capteur ou de l'environnement lors de l'acquisition, Ils suppriment donc les détails tout en conservant les contours. Alors que les hapax dans nos images représente le contour , ce la veux dire que la conversion des hapax nous amène au pouvoir discriminant de classification

5. Interface de SYCLIM et mode d'utilisation

Avec SYCLIM l'utilisateur peut manipuler l'interface et faire fonctionner les différentes tâches avec une grande souplesse, il est facile à utiliser. Les différentes étapes à suivre pour arriver à la phase finale de classification sont comme suit :

- *Le bouton Ouvrir*

Permet d'ouvrir une boîte de dialogue qui permet de sélectionner les différentes images.

- *Le bouton Conversion*

Il a pour rôle de convertir les images couleurs aux niveaux de gris selon la formule décrite précédemment. Si ces images sont déjà aux niveaux de gris lors de leur conversion, un message apparaîtra en indiquant que l'image est déjà aux niveaux de gris.

- *Le bouton lissage*

Après la conversion de l'image, cette dernière peut subir un filtrage pour éliminer les bruits.

- *La fonction N-grams*

L'utilisateur a le choix entre les bigrams et les trigrams. Cette étape vient après les phases de conversion et lissage. Pour chacun des deux choix la matrice vectorielle des images sera construite, en prenant compte des trois paramètres (le numéro de l'image ou segment, les n-grams et leurs fréquences).

• *Le bouton Classification*

Cette tache, lance le processus de classification, en appelant notre classifieur qui recevra les données, en input, représentées par la matrice des N-grams.

Les figures qui viennent montrent les différentes étapes décrites avec des interfaces graphiques (voir les figures de V.8 à V.12).

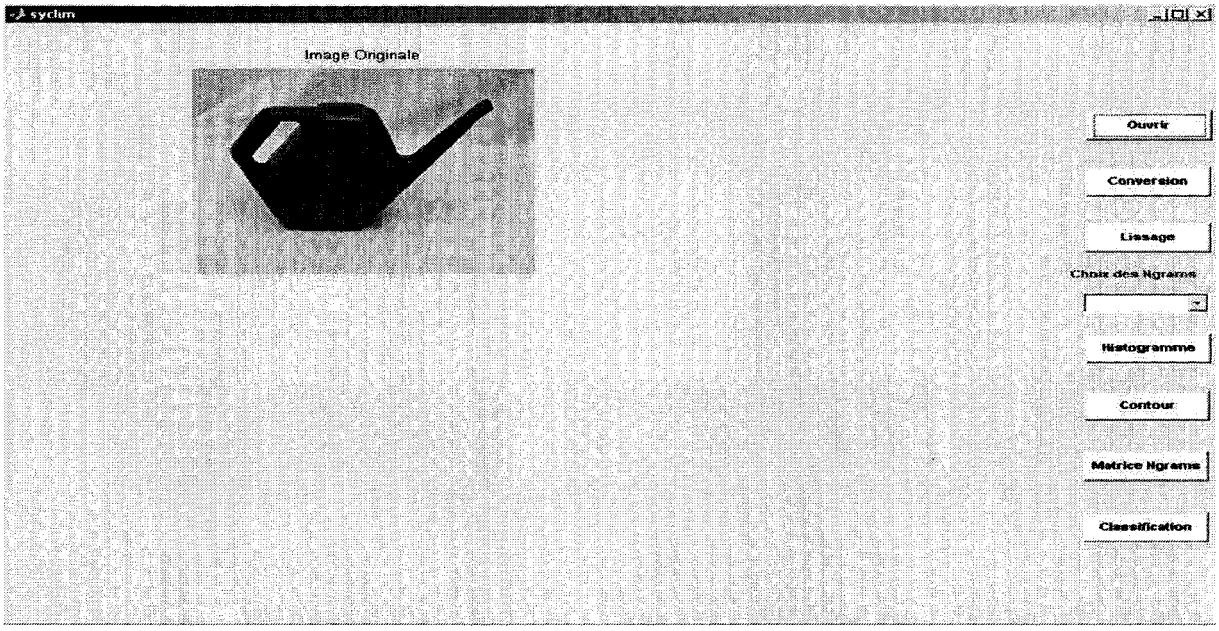


Figure V.8 : Acquisition d'images

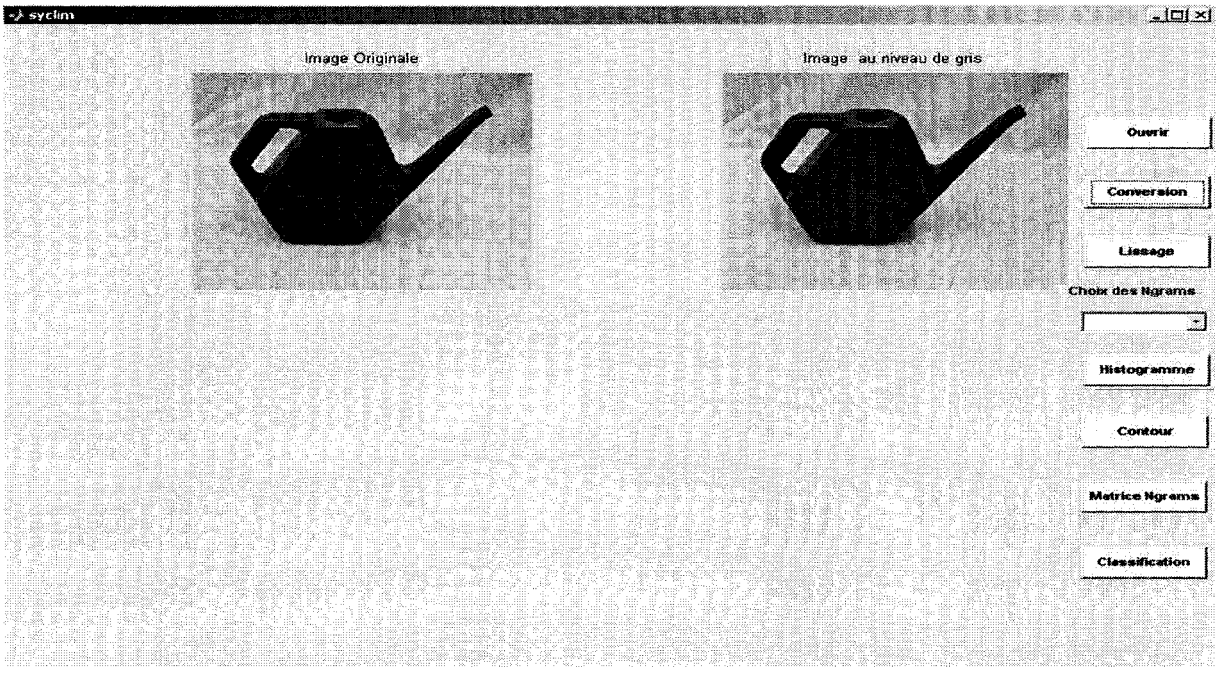


Figure V.9 : Conversion aux niveaux de gris.

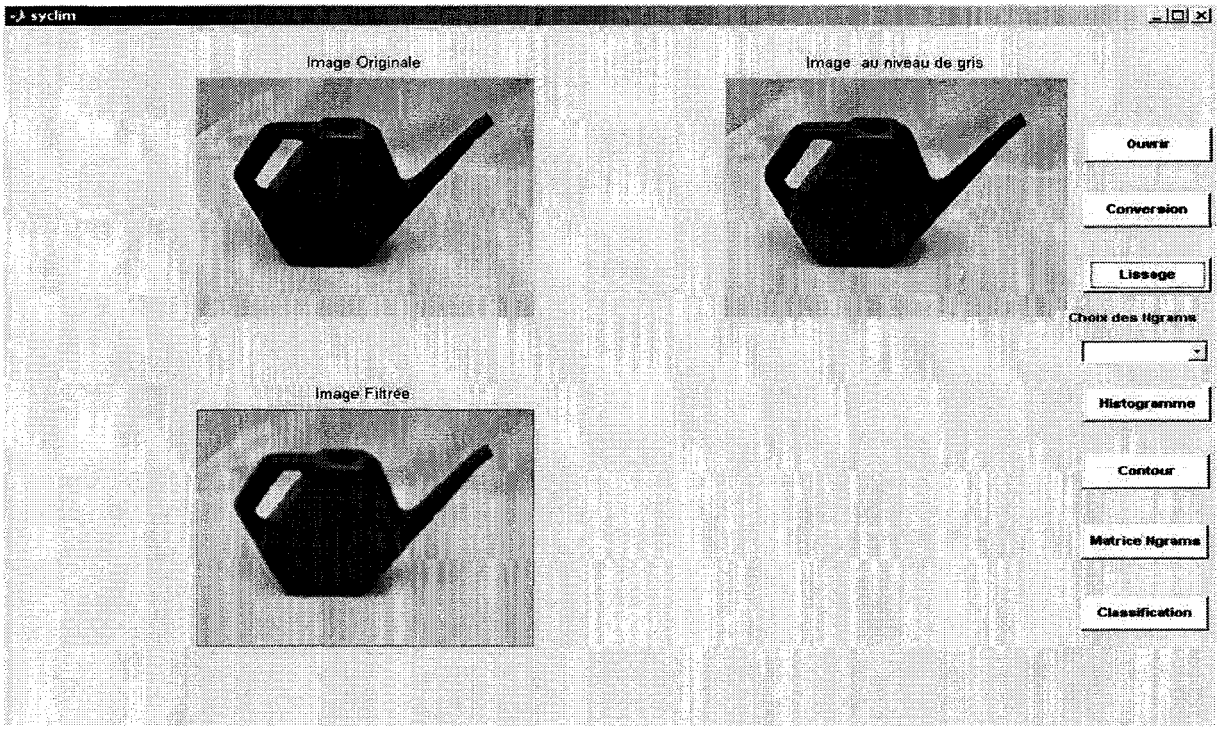


Figure V.10 : Filtrage des images.

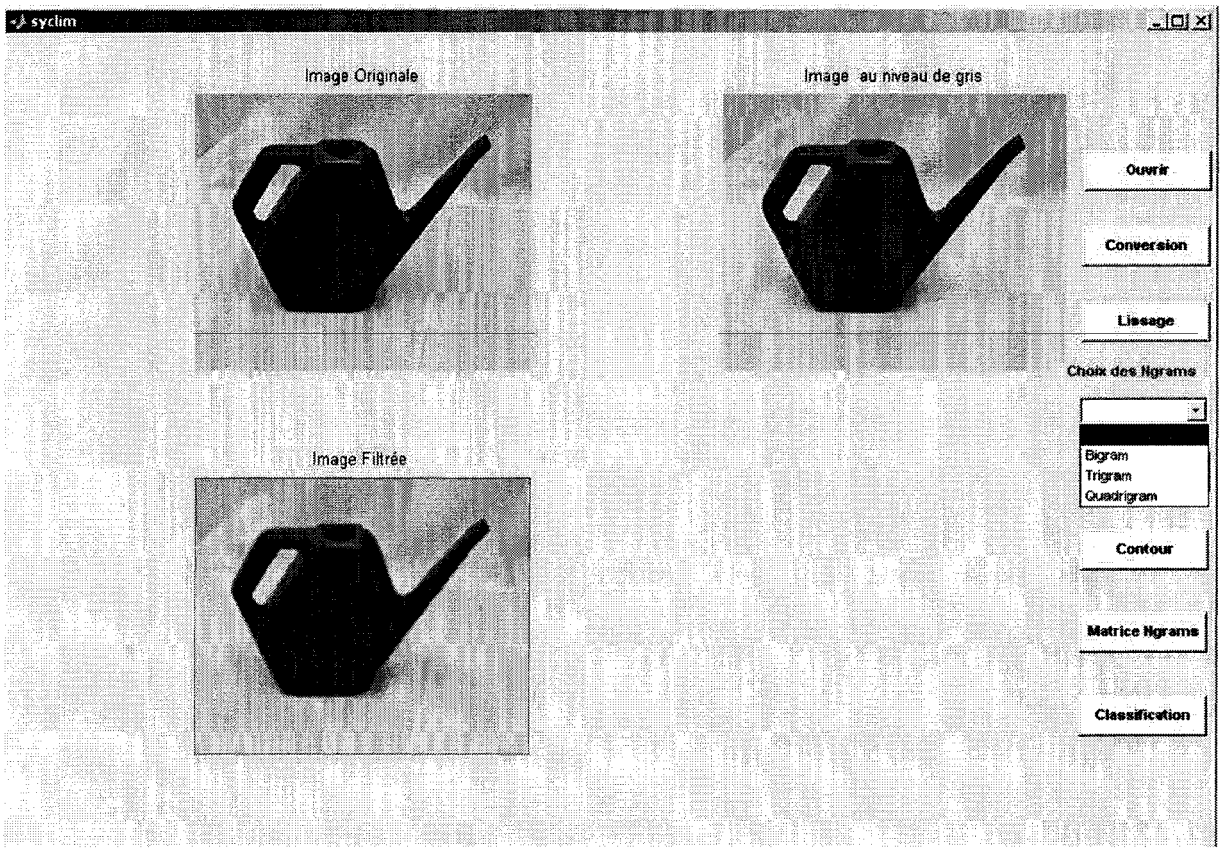


Figure V.11 : Choix des NGVIP.

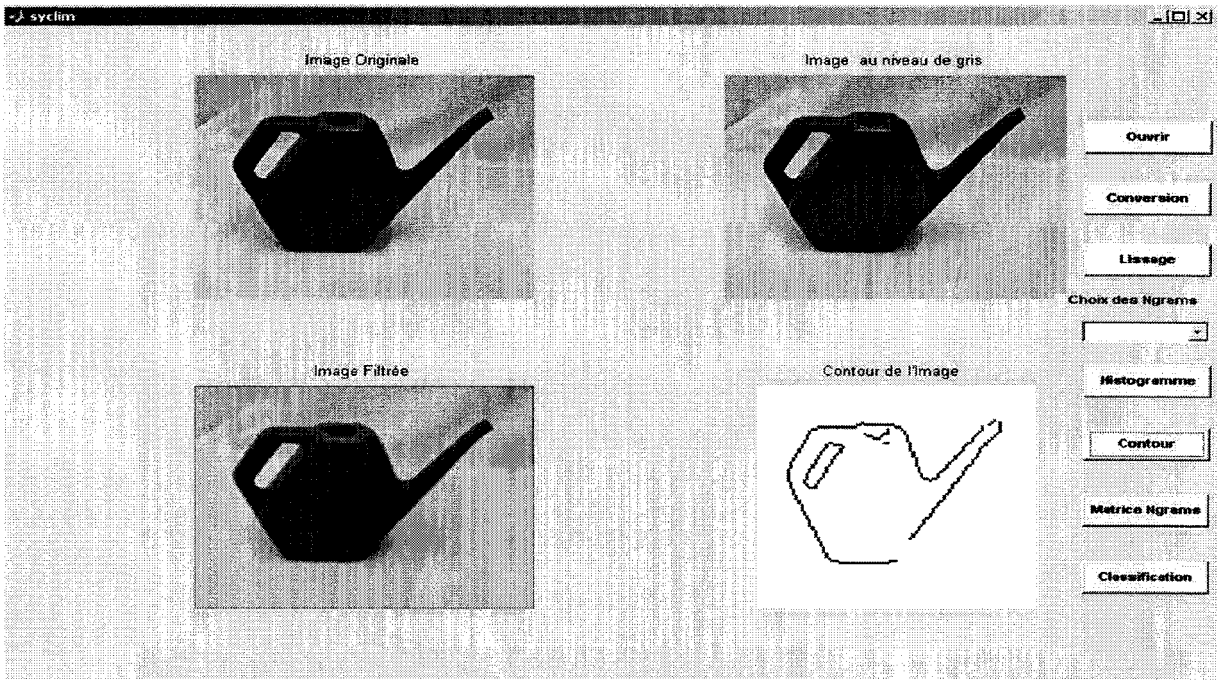


Figure V.12 : Calcul du contour de l'image.

Le tableau V.5, montre comment nos images sont présentées vectoriellement, on présente les NGVIP des images et leur fréquence d'apparition. L'exemple considéré consiste en un découpage en bigrams.

Bigrams	Segment	Fréquence
1	3	23
1	4	23
1	15	69
1	17	2
1	22	2
2	3	7
2	4	7
2	9	1
2	15	19
2	16	3
2	17	4
2	22	1
2	23	1
3	3	9
3	4	9
3	15	7
3	16	4
3	17	3
3	22	3
4	3	2
4	4	2
4	9	1
4	10	1
4	11	1
4	12	1
4	13	1
4	15	2
4	16	2
4	17	2

Tableau V.8 : La matrice vectorielle des images

CHAPITRE
VI

Conclusion et Perspectives

Conclusion

Dans ce mémoire, nous avons proposé une méthode semi automatique de classification pour les images numériques, l'approche ayant déjà fait ses preuves dans la classification textuelle et le traitement de la parole. La méthode proposée est celle des N-Grams. Cette dernière est basée sur les valeurs d'intensités des pixels et plus particulièrement les combinaisons adjacentes de ces valeurs d'intensités et leurs fréquences d'apparitions. Ces deux informations forment le noyau d'informations pour notre classifieur à titre de données d'entrées du classifieur.

À cet égard, SYCLIM (Système de Classification des Images) est le prototype que nous avons développé pour classifier les images de notre base de données. Cette base, représente des images des personnes, des paysages, des objets, etc. Elles sont toutes au format JPEG avec une résolution de 320*200. Certaines sont en couleur, autres en niveaux de gris. Ces images peuvent subir plusieurs pré-traitements, tel que la conversion des images couleurs en niveaux de gris afin d'avoir une représentation bidimensionneflle commune, ainsi que le filtrage (lissage) pour la diminution du bruit.

L'étape qui intervient après les phases de pré-traitements consiste en l'extraction des N-Grams de nos images avec des représentations en BGVIP (Bigrams de valeurs d'intensités des pixels) ou en TGVIP (Trigrams de valeurs d'intensités des pixels). Une représentation vectorielle est réalisée pour nos images structurées par les indices des images qui représentent les numéros des chacune d'elles, par les N-Grams et par les fréquences d'apparitions correspondantes.

Nous justifions notre choix par la nécessité d'avoir une approche plus générale pour classifier toutes les sources d'informations, peu importe leur nature. Ce qui est très approprié

étant données les contraintes que nous rencontrons de nos jours avec l'Internet. En effet, le concept des N-Grams est indépendant de la nature et de l'interprétation sémantique de l'information. Il tient compte seulement du caractère (dans sa plus large définition) utile pour représenter les données (Pixel pour une image ou caractère ASCII pour le texte).

Des résultats très significatifs ont été obtenus. Cette approche informatique très économique est adaptable aux images. On note ce qui suit :
(i) Les classes sont plus cohérentes quand elles sont le résultat d'une classification sur des images du niveau tridimensionnel en couleur sans éliminations du hapax.
(ii) Augmentations du niveau discriminant quand nous choisissons des découpages en trigrams (TGVIP) au lieu du découpage en bigrams (BGVIP).

Perspectives

Dans les travaux futurs, il y a plusieurs voix intéressantes à explorer. Plusieurs évaluations peuvent être effectuées dans plusieurs domaines en vision pas ordinateur, en pattern classification et l'indexation d'images. Ce travail sera une étape significative pour notre objectif final qui est la classification de l'information multimédia quelque soit sa nature et la recherche d'information sur Internet, et puisque le concept des N-grams est indépendant de la nature et de la sémantique de l'information préconisées.

- La classification des documents là ou le texte et l'image sont tous deux présents.

- L'indexation et la recherche d'images, ce qui permet de rechercher des images similaires à une requête dans une base d'images en se basant sur les caractéristiques propres aux images. En effet, des mesures peuvent être calculées par notre approche et qui permettent de calculer la distance entre deux images, image requête et image de la base d'images. Ce qui donne comme résultats non pas une classe d'appartenance, mais un certain nombre d'images jugées pertinentes et similaires à l'image requête proposée.

- L'indexation multimédia, l'archivage, la recherche d'information dans des documents sonores, l'indexation audio et identification de méta-données comme le cas de transcription d'émissions

radio ou télédiffusées, indexation multimédia pour la recherche d'information dans des documents sonores, identification de méta-données (locuteur, langue...).

Bibliographie

[ALE 03] Alexander Kolesnikov , Pasi Fränti , *" Reduced-search dynamic programming for approximation of polygonal curves"*. Pattern Recognition Letters 24 (2003) 2243–2254.

[BAR 02] Barequet G, Chen .D.Z, Daescu .O, Goodrich .M.T, Snoeyink .J, *" Efficiently approximating polygonal paths in three and higher dimensions"*. Algorithmica, 33(2): 150-167, 2002.

[BIS 01] Biskri .I, Delisle .S, *" Les n-grams de caractères pour l'extraction de connaissances dans des bases de données textuelles multilingues"*. TALN 2001, Tours, France.

[BIS 02] Biskri .I, Delisle .S, *" Text Classification and Multilinguism: Getting at Words via N-grams of Characters"*. Proceedings of the 6th World Multi-Conference on Systemic, Cybernetics and Informatics (SCI'02) & the 8th International Conference on Information Systems, Analysis and Synthesis (ISAS'02), Orlando, États-Unis, 2002.

[BOR 02] Borgefors .G, Svensson .S, *" Fuzzy border distance transforms and their use in 2D skeletonization"*. Proc. Int. Conf. on Pattern Recognition-ICPR'02, Quebec City, Canada, 1: 180 –183, August 2002.

[BOU 03] Bouloudani .N, Lambert .P, Coquin .D, *" Segmentation automatique des images couleur à base d'indicateurs de performance"*. CORESA'03, 2003.

[CAV 94] Cavnar .W and Trenkl .J, *" N-Gram Based Text Categorization"*. In *Symposium on Document Analysis and Information Retrieval*, Las Vegas, 1994.

[CEL 89] Celeux .G, Diday .E, Govaert .G, *" Classification automatique de données environnement statistique et informatique"*. Dunod, Informatique, 1989.

[CHA 94] Chateau .F, *" Probabilités a priori inégales dans la règle des k plus proches voisins"*. Actes des XXVIèmes Journées de Statistiques (Neuchâtel):195-198, 1994.

[CHU 02] Chung .K.L, Yan .W.M, Chen W.Y, *" Efficient algorithms for 3-D polygonal approximation based on LISE criterion"*. Pattern Recognition, 35: 2539-2548, 2002.

[DAM 95] Damashek .M, " *Gauging Similarity with N-grams: Language-Independent Categorization of Text*". Science, (267):843–848, 1995.

[DAS 91] Dasarathy .B.V, *" Nearest Neighbour (NN) norms: NN Pattern Classification Technique"*. IEEE Computer Society Press, Los Alamitos, CA, 1991.

[DES 98] Desseilligny .M.P, Stamon .G, Suen .Ch.Y, *" Veinerization: A New Shape Description for Flexoble Sceletonization*". IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 20, No. 5,pp. 505-521, 1998.

[DID 72] Diday .E, *" Optimisation en classification automatique et reconnaissance de formes"*. Note Scient. IRIA n° 6, 1972.

[DOR 99] Dori Dov and Wenyin Liu, *" Sparse Pixel Vectorization: An Algorithm and Its Performance Evaluation"*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 21, No.3, pp.202-215, 1999.

[DUN 94] Dunning T, *" Statistical Identification of Languages"*. Technical Report MCCS 94-273, Computing Research Laboratory, New Mexico State University, 1994.

[FRA 02] Fränti .P, Ageenko .E, Kukkonen .P, Kälviäinen .H, *" Using Hough transformation for context-based image compression in hybrid raster/vector applications"*. Journal Electronic Imaging, 11(2): 236-245, 2002.

[GRE 95] Greffenstette .G, *" Comparing Two Language Identification Schemes"*. *Actes de JADT-95*, 85-96, 1995, France.

[GRI 03] Gribov .A, Bodansky .E, *" Vectorization with the Voronoi L-diagram"*. The Seventh International Conference on Document Analysis and Recognition (ICDAR 2003).

[HAL 98] Halleb .M, Lelu .A, *" Hypertextualisation Automatique Multilingue à Partir des Fréquences de n-Grammes"*, *Actes de JADT-98*, Nice, France, 1998.

[HUF 94] Huffman .S, Damashek .M, *" Acquaintance: A novel vector-space n-gram technique for document categorization"*. In NIST Special Publication 500-226: Overview of the Third Text Retrieval Conference (TREC-3), 1994, 305-310, Washington, D.C, 1994.

[JIQ 02] Jiqiang Song, Feng Su, Chiew-Lan Tai, Shijie Cai, *" An Object-Oriented Progressive-Simplification-Based Vectorization System for Engineering Drawings: Model, Algorithm, and Performanc"*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1048 - 1060, Volume 24, 2002.

[JIQ 00] Jiqiang Song, Feng Su, Jibing Chen, Chiewlan Tai, Shijie Cai, *" Line Net Global Vectorization: an Algorithm and Its Performance Evaluation"*. Computer Vision and Pattern recognition (CVPR'00)-Vol1, P1383, Hilton Head, South Carolina, 2000.

[JOS 98] José Martinez and Sylvie Guillaume, *" Colour image retrieval fitted to classical querying"*. In Network and Information Systems Journal (NJIS), 1998.

[JOS 02] José Martinez and Erwan Loisant, *" Browsing image databases with Galois' lattices"*. In Proceedings of SAC'2002, 2002.

[KAR 00] Karl Tombre, Salvatore Tabbone, *" Vectorization in Graphics Recognition: To Thin or not to Thin"*. 15th international conference on pattern recognition, September 2000, Barcelona , Spain.

[KOL 03] Kolesnikov .A, Fränti .P, *" Data Reduction of Large Vector Graphics"*. Research Report, A-2003-2, CS Dept, University of Joensuu, Joensuu, Finland, 2003.

[KOS 98] Kossentini .F, Smith .M.J.T, *" A fast PNN design algorithm for entropy constrained residual vector quantization"*. IEEE Trans. Image Processing, 7:1045-1050, 1998.

[KRO 92] Krovetz .R & Croft .W, *" Lexical ambiguity and information retrieval. Transactions on Information Systems (TOIS)"*. Publication of the Association for Computing Machinery (ACM), 10 (2), 115–141, 1992.

[LEE 99] Lee .M.L, Lu .H, Ling .T.W, Ko .Y.T, *" Cleansing Data for Mining and Warehousing"*. Proceeding of 10th International. Conference on Database and Expert Systems Applications (DEXA), 1999.

[LEL 98] Lelu A., Halleb M. , Delprat B, *" Recherche d'information et Cartographie dans des Corpus Textuels à Partir des Fréquences de n-Grammes"*, *Actes de JADT-98*, Nice, France, 1998.

[MAN 03] Manzanera .A, Bernard .T.M, *" Metrical properties of a collection of 2D parallel thinning algorithms"*. Proc. Int. Workshop on Combinatorial Image Analysis, Palermo, Italy, May 2003, Electronic Notes on Discrete Mathematics, vol. 12, Elsevier Science, 2003.

[MAY 98] Mayfield .J, Mcnamee .P, *" Indexing Using both n-Grams and Words"*. *NIST Special Publication 500-242 : TREC 7*, 419-424, 1998.

[MCL 92] Mclachlan .G.J, *" Discriminant Analysis and statistical Pattern Recognition"*. Wiley et Sons, Inc, 1992.

[MEU 97] Meunier .J.G, Biskri .I, Nault .G, Nyongwa .M, *" Exploration de classifieurs connexionnistes pour l'analyse terminologique"*. Colloque RIAO97, Montréal, pages 661-664, 1997.

[MEU 02] Meunier .J.G, Biskri .I, *" SATIM : une plate-forme modulaire pour la construction de chaînes d'analyse de textes assistée par ordinateur"*. (Articles sélectionnés du Colloque international : L'édition électronique en littérature et dictionnairique : évaluation et bilan. Rouen, France, Juin 2002.). Dans L'édition électronique : état des lieux. Presses de L'Université de Rouen, J.C. Arnould, Claude Blum. (Eds). 22 pages.

[MIK 01] Mikheev .A, Vincent .L, Faber .V, *" High-quality polygonal contour approximation based on relaxation"*. Proc. Int. Conf. Document Analysis and Recognition-ICDAR'01, 361-365, 2001.

[MIL 99] Miller .E, Shen .D, Liu .J and C. Nicholas, *" Performance and Scalability of a Large-Scale N-gram Based Information Retrieval System"*. Journal of Digital Information, Volume 1, Issue 5, No. 21, 1999.

[MUK 01] Mukesh Kumar and D.Surya Srinivas, *" Unsupervised Image Classification By Radial Basis Function Neural Networks(RBFNN) "*. 22nd Asian Conference on Remote Sensing, November 2001, Singapore.

[OTS 79] Otsu. N, *" A Threshold Selection Method From Gray Level Histograms "*. IEEE Trans. Syst.Man Cyber., 1979, Vol. 9, No. 1, pp. 62-66.

[PER 98] Perny, .P, *" Multicriteria filtering methods based on concordance and non-discordance principles"*. Annals of Operations Research 80:137-165, 1998.

[QIN 03] Qing Wu and Yizhou Yu, *" Two-Level Image Segmentation Based on Region and Edge Integration"*. Proc. VIIth Digital Image Computing: Techniques and Applications,Dec. 2003, Sydney.

[QUI 93] Quinlan .J.R, *" Programs for Machine Learning"*. Morgan Kaufmann publishers, San Mateo, California, 1993.

[RES 97] Resnik .P & Yarowsky .D, *" A perspective on word sense disambiguation methods and their evaluation"*. Association for Computational Linguistics Special Interest Group on the Lexicon". (ACL-SIGLEX-1997) : Workshop « Tagging Text with Lexical Semantics : Why, What, and How? », 79–86, 1997.

[REV 01] Revollon .P, Foucher .P, Boumaza .R, *" Étude Comparée de Trois Techniques de Segmentation d'images de Végétaux : seuillage d'histogramme, analyse discriminante et réseaux de neurones"*. XXIIIèmes Journées de Statistiques, ENITIAA, Nantes, 14-18 mai 2001.

[RIC 03] Richard Lepage, Bassel Solaiman, *" Les Réseaux de Neurones artificiels et Leurs applications en Imagerie et en Vision Par Ordinateur"*. École de technologie Supérieure , Québec 2003.

[RUM 86] Rumelhart .D, Hinton .G, Williams .R, *" Learning internal representations by error propagation"*. In: parallel distributed processing: explorations in the microstructure of cognition. Eds Cambridge, MA: MIT Press, 1986.

[SAB 04] Sabrina Tollari, Herve Glotin, Jacques Le Maitre, *" Rehaussement de la classification textuelle d'images par leur contenu visuel"*. Laboratoire SIS - Equipe informatique Université de Toulon, 2004, France.

[SAH 99] Sahami .M, *"Using Machine Learning to Improve Information Access*. Ph.d. thesis, Computer Science Department, Stanford University, 1999.

[SAL 02] Salotti .M, *" Un algorithme efficace pour l'approximation polygonale optimale"*. Proc. 13ème Congrès Francophone AFRIF-AFIA de Reconnaissance des Formes et Intelligence Artificielle-RFIA'2002, Angers, France, 1: 11-18, 2002.

[SAN 94] Sanderson .M, *" Word sens disembiguation and information retrieval. Association for Computing Machinery Special Interest Group on Information Retrieval"*. (ACM-SIGIR-1994):

17th Annual International Conference on Research and Development in Information Retrieval, 142–151, 1994.

[SON 02] Song .J, Su .F, Tai .C.L, Cai .S, *" An object-oriented progressive simplification based vectorization system for engineering drawings: Model, algorithm, and performance"*. IEEE Trans Pattern Analysis Machine Intelligence, 24(8): 1048-1060, 2002.

[SON 00] Song .J, Su .F, Chen .J, Cai .S, *" A knowledge-aided line network oriented vectorization method for engineering drawings"*. Pattern Analysis and Applications, 3: 142-152, 2000.

[SPA 80] Späth .H, *" Cluster Analysis Algorithms for data reduction and classification of objects"*. Ellis Horwood, Willy & Sons, New York, 1980.

[TEY 01] Teytaud .O and Jalam .R, *" Kernel-based text categorization"*. In *IJCNN'01*, Washington, DC, USA, 2001.

[THI 01] Thierry Géraud, Pierre-Yves Strub, Jérôme Darbon, *" Segmentation d'Images en Couleur par Classification Morphologique Non Supervisée"*. International Conference on Image and Signal Processing (ICISP'2001), Agadir, Morocco, May 2001.

[YAN 00] Yang C.C, Prasher S.O, Landry J.A, Ramaswamy H.S. and. Ditommaso .A, *" Application of artificial neural networks in image recognition and classification of crop and weeds"*. Canadian Agricultural Engineering Vol. 42, No. 3, 2000.

[YAR 94] Yarowsky .D, *" A comparision of corpus-based techniques for restoring accents in spanish and french text"*. 2nd Annual Workshop on Very Large Text Corpora, 19–32, 1994.

[YAR 94] Yarowsky .D, *" Decision lists for lexical ambiguity resolution : Application to accent restoration in spanish and french"*. 32nd Annual Meeting of the Association for Computational Linguistics (ACL-1994), 88–95, 1994.