

M0014/03A

République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

ECOLE NATIONALE POLYTECHNIQUE

Département d'Electronique

Thèse de Magister

Présentée par :

المدرسة الوطنية المتعددة التقنيات
المكتبة —
BIBLIOTHEQUE —
Ecole Nationale Polytechnique

M^{me} DIAF Fatiha

Ingénieur U.S.T.O.

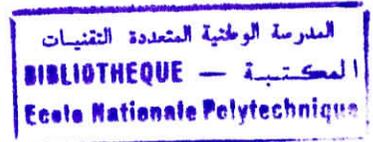
Pour l'obtention du grade de **MAGISTER** en Electronique
Option : **Image et Parole**

RECONNAISSANCE AUTOMATIQUE DE FORMES **Application au tri automatique de graines de semences**

Soutenue publiquement le 27 septembre 2003 devant le jury composé de :

Président	: HADDADI M.	Maître de Conférences	E.N.P.
Rapporteur	: GUERTI Mhenia	Maître de Conférences	E.N.P.
Examineur	: ZELMAT M.	Maître de Conférences	Univ. Boumerdès
Examineur	: HAMMAMI L.	Docteur d'Etat	E.N.P.

AVANT-PROPOS



Cette thèse a été dirigée par Mlle **GUERTI Mhenia**, Maître de Conférences à l'Ecole Nationale Polytechnique. Nous lui adressons nos plus vifs remerciements pour son aide et pour ses encouragements constants.

Nos vifs remerciements vont également à Monsieur **M. HADDADI**, Maître de Conférences à l'Ecole Nationale Polytechnique., pour nous avoir fait honneur de présider le jury de cette thèse.

Nous sommes très sensibles à l'intérêt que Monsieur **M. ZELMAT**, Maître de Conférences à l'Université de Boumerdès porte à ce travail et nous lui exprimons nos plus vifs remerciements pour sa participation à notre jury de cette thèse.

Que Mme **L. HAMMAMI**, Docteur d'Etat, Enseignante à l'Ecole Nationale Polytechnique, trouve ici l'expression de notre profonde gratitude pour avoir bien voulu examiner ce travail en acceptant de faire partie du jury.

ملخص

يعتبر الهدف من هذا العمل دراسة و تحقيق طريقتين لتعريف الأشكال. نعتبر في الطريقة الأولى بعض الوسائط المرفولوجية من صور الأشكال التي نريد معرفتها. نستعمل في الطريقة الثانية تحاليل عوامل النسيج التي تحسب من المصفوفة الخاصة بشبكة الخلايا متعددة الطبقات. إن تطبيق هذين الطريقتين أعطى نتائج هامة.
كلمات جوهرية: تعريف الأشكال، شبكات الخلايا

Résumé

Le but de notre travail est l'étude et la réalisation de deux méthodes de Reconnaissance Automatiques de Formes. Dans la première méthode, nous extrayons un certain nombre de paramètres morphologiques à partir des images des échantillons des objets à reconnaître. Ensuite, après l'application de l'Analyse en Composantes Principales (A.C.P.) suivie de la Classification Automatique basée sur l'agrégation Autour des Centres Mobiles aux données morphologiques obtenues, la signature polaire moyenne de chaque classe en présence est déterminée. La reconnaissance d'un objet s'effectue alors, en l'affectant à la classe correspondant à la plus petite erreur quadratique entre sa propre signature polaire et les signatures polaires moyennes de toutes les classes. En seconde partie, la méthode proposée est basée sur l'analyse de paramètres de texture calculés à partir de la matrice de cooccurrence et d'un réseau de neurones multicouches utilisé pour la classification dont l'apprentissage est réalisé par l'algorithme de rétropropagation. Appliquées à un échantillon de quatre cents graines de quatre espèces différentes, les deux méthodes ont fourni des résultats très intéressants. Mots clés : Reconnaissance des Formes, Réseaux de Neurones, Classification Automatique, Tri de graines, A.C.P.

Abstract

The aim of this work is the study and the computing of three pattern recognition methods. In the first method, morphological features are extracted from the objets to recognise. After the application of the Principal Component Analysis followed by the K-means method, the middle of the polar signatures of each class is determined. The decision about an unknown object is made by affecting it to the class corresponding to the smallest means quadratic error between its polar signature and the middle polar signature of the classes. In the second part of this work, the second method proposed is based o the texture feature consideration. This features are calculated from the cooccurrence matrix. A multilayer neural network is used. The backpropagation algorithm is used in learning. The results obtained with the two methods are discussed ad compared.

Key-words : Pattern Recognition, Images Analysis, Classification, grains select, Neural Network.

Introduction Générale.....i

CHAPITRE 1 : GENERALITES SUR LA RECONNAISSANCE DES FORMES.

1.1 – Introduction	4
1.2 – Principe de la reconnaissance des formes.....	5
1.3 – Normalisation et Réduction des paramètres.....	6
1.3.1 Normalisation des paramètres.....	6
1.3.2 Réduction des paramètres.....	6
1.4 – Classification Automatique.....	8
1.4.1. Décision bayésienne.....	8
1.4.2. Découpage en fenêtres de Parzen.....	9
1.4.3 Classification par les k plus proches voisins.....	9
1.4.4 Méthodes hiérarchiques.....	10
1.4.5 Méthode d'agrégation autour des centres mobiles.....	11
1.4.6 Méthode mixte.....	11
1.5 – Conclusion.....	13

CHAPITRE 2 : TRAITEMENT D'IMAGES EN RECONNAISSANCE DE FORMES

2.1 Introduction.....	15
2.2 Acquisition des images des graines.....	15
2.3 Images en couleur	16
2.3.1 Système de représentation de la couleur.....	17
2.3.2 Paramètres de couleur.....	18
2.4. Images texturées.....	20
2.5 Filtrage d'images.....	23
2.6 Binarisation d'images.....	27
2.6.1 Méthodes de binarisation par seuillage global.....	28
2.6.1.1 Seuillage par maximisation d'entropie.....	28
2.6.1.2 Seuillage par maximisation de la variance.....	29
2.6.1.3 Seuillage par maximisation de l'erreur quadratique	30
2.6.1.4 Seuillage par conservation des moments.....	31
2.6.2 Méthodes de binarisation par seuillage local.....	31
2.6.2.1 Seuillage par test de validation.....	32
2.6.2.2 Seuillage par maximisation d'un critère d'homogénéité.....	33
2.6.3 Comparaison entre les méthodes de binarisation.....	34
2.6.3.1 Critère d'uniformité.....	35
2.6.3.2 Critère de contour.....	35
2.6.3.3 Résultats obtenus.....	35
2.7 Suivi de contours.....	37
2.8 Conclusion.....	39

CHAPITRE 3 : RECONNAISSANCE DE GRAINES A PARTIR DES PARAMETRES MORPHOLOGIQUES

3.1 – Introduction.....	41
3.2 – Acquisition des images.....	41
3.3 – Binarisation des images.....	42
3.4 – Suivi de contours.....	43
3.5 – Calcul des paramètres morphologiques.....	43
3.6 – Normalisation et réduction des paramètres morphologiques.....	46
3.7 – Classification automatique des données.....	48
3.7.1 Association avec l'ACP.....	48
3.7.2 Utilisation de la méthode mixte.....	48
3.7.3 Résultats obtenus.....	49
3.8.-Signature polaire.....	51
3.8.1 Calcul de la signature polaire.....	51
3.8.2 Modèle d'apprentissage.....	51
3.9.-Phase de reconnaissance.....	53
3.10 Logiciel.....	54
3.11 Conclusion.....	56

CHAPITRE 4 : RECONNAISSANCE DE GRAINES A L'AIDE DES PARAMETRES DE TEXTURE ET DE RESEAUX DE NEURONES

4.1 – Introduction.....	58
4.2 – Rappel sur les réseaux de neurones.....	58
4.2.1 Neurone formel et fonction d'activation.....	59
4.2.2 Réseaux statiques ou non bouclés.....	60
4.2.3 Réseaux dynamiques ou bouclés.....	62
4.2.4 Apprentissage.....	63
4.2.5 Réseaux à couches.....	64
4.2.6 Algorithme de rétropropagation.....	66
4.3. – Mise au point d'un R.N.....	68
4.4 Application des R.N. à la reconnaissance des graines.....	69
4.4.1 Principe.....	69
4.4.2 Prétraitement des images.....	70
4.4.3 Extraction des paramètres de texture.....	70
4.4.4 Apprentissage.....	72
4.5 Tests et Résultats.....	74
4.6 Conclusion.....	74

CONCLUSION GÉNÉRALE

BIBLIOGRAPHIE

Introduction Générale

Parmi les facultés intellectuelles de l'homme, la plus importante est certainement la perception, c'est-à-dire l'acquisition, l'évaluation et l'interprétation de données sensorielles. Malgré la simplicité apparente de cette démarche intellectuelle, il est actuellement impossible de connaître le processus mis en jeu au niveau du système nerveux. Il est cependant possible d'aboutir à des résultats semblables en utilisant des techniques diverses de la théorie de l'information et de la théorie de la décision. Ces disciplines très liées à l'Automatique et à l'Informatique industrielle conduisent au traitement et à l'interprétation automatique des données issues de l'environnement étudié.

Dans ce contexte, notre intérêt s'est particulièrement porté sur la Reconnaissance Automatique de Formes (RAF). Si le schéma de principe de la RAF est commun à tous les domaines appliqués à cette discipline, d'une application à l'autre, les techniques introduites sont souvent différentes. Lorsqu'il s'agit, par exemple, de trier des pièces de formes géométriques classiques, la reconnaissance peut être effectuée sur la base de règles syntaxiques en utilisant les dimensions géométriques. Cependant, lorsqu'il s'agit de trier des objets dont les morphologies s'approchent d'une espèce à l'autre, le problème devient compliqué. Il y a lieu de mettre au point de nouvelles méthodes de classification automatique performantes par rapport aux méthodes classiques pour pouvoir séparer les classes qui se chevauchent. Par ailleurs, si les paramètres morphologiques n'offrent pas un taux de reconnaissance suffisant, d'autres paramètres liés à la couleur et à la texture peuvent être introduits.

L'application qui fait l'objet de cette thèse est une contribution au tri automatique de graines de semences qui constituent des objets porteurs

d'informations très riches en matière particulièrement de forme et de texture. Ce projet pourra être utilisé par :

- des agronomes pour sélectionner des espèces à partir des caractéristiques de leurs graines,
- Des contrôleurs de qualité de chargements de céréales importées afin de calculer automatiquement le taux ou l'indice des impuretés contenues dans ces chargements.

Dans le cas d'importation des graines de céréales, l'opération habituelle est manuelle. Elle consiste à prélever un échantillon du chargement pour l'étudier en s'appuyant sur des documents photographiques. Cette opération est fatigante et prend beaucoup de temps. Elle comporte des risques sur la santé et les troubles respiratoires dus aux poussières résultant des traitements chimiques de ces graines avec des pesticides souvent à base de sels de mercure. Ceci nous a amenés à l'automatisation de la procédure en utilisant une caméra pour saisir les images des graines.

Ainsi, dans le premier chapitre, nous présentons les généralités sur la reconnaissance des formes, le principe, les domaines d'application et la classification automatique.

Le second chapitre est consacré aux techniques de traitement d'images qui interviennent dans le prétraitement tels que le filtrage, la binarisation, les suivis de contours et la texture.

L'application proprement dite de reconnaissance à partir des paramètres de forme des graines constituera le troisième chapitre où seront présentées les méthodes utilisées et les résultats obtenus.

Dans le quatrième chapitre, nous utiliserons les paramètres de texture et des nouvelles méthodes performantes de classification automatique sortant du cadre des méthodes classiques habituellement utilisées telles que les réseaux de neurones pour reconnaître les graines.

1.1 Introduction

La reconnaissance de formes consiste à affecter une forme à la classe à laquelle elle appartient. Elle est définie comme étant l'ensemble des techniques qui permettent de reproduire la perception humaine en vue d'automatiser complètement les opérations et les prises de décision [1,2]. En ce qui concerne notre travail, c'est la reconnaissance des formes par vision artificielle qui nous intéresse. Les données sont donc des images d'objets saisies par caméra. Ceci nécessite la maîtrise de certaines techniques de traitement d'images et de classification automatique. Conserver en mémoire de l'ordinateur les images de tous les objets pour ne réduire la reconnaissance qu'à une simple comparaison nécessiterait un espace mémoire impossible à satisfaire. On préfère alors décrire chaque objet O_i par un certain nombre de paramètres ayant m composantes notées x_{ij} ($i=1,2,3,\dots,n$ et $j=1,2,3,\dots,m$) qui leurs sont très caractéristiques :

$$X_{ij} = (x_{i1}, x_{i2}, \dots, x_{ij}, \dots, x_{im}), i = 1, 2, \dots, n \quad (1)$$

$$j = 1, 2, \dots, m$$

C'est l'ensemble des m composantes de ce vecteur qui constitue l'espace de représentation. Ces données sont rangées sous forme d'un tableau objet/paramètre (tableau 1).

Objet/Paramètre	p_1	p_2	p_j	p_m
O_1	x_{11}	x_{12}	x_{1j}	x_{1m}
O_2	x_{21}	x_{22}	x_{2j}	x_{2m}
..
O_i	x_{i1}	x_{i2}	x_{ij}	x_{im}
..
O_n	x_{n1}	x_{n2}	x_{nj}	x_{nm}

Tableau 1 : Objet/Paramètre

La nature de ces données très diversifiées diffère selon le domaine d'application (tableau 2).

Domaine d'étude	Application	Capteur
Reconnaissance de la parole (mots isolés, monolocuteur, multilocuteurs, discours)	Commande vocale, Bureautique Identification d'individus...	Microphone
Lecture automatique	saisie de textes, tri postal...	Lecteur optique scanner, barrettes CCD
Diagnostic médical	analyse des signaux électro-physiologiques, comptage de globules, reconnaissance de chromosomes,...	électrodes, échographies, caméra
Reconnaissance de signatures	banques, commerce,...	caméra, lecteur optique
Reconnaissance d'empreintes digitales	banques, commerce, polices	caméra, scanners
Inspection automatique	détection de défauts dans les textiles, les objets industriels...	barrettes CCD, caméra, système à balayage électrique
Identification d'objets	tri d'objets, surveillance	caméra, radar, barrettes CCD
Localisation d'objets	navigation de robots, guidage de missiles,...	capteurs de proximité, caméra, télémètre
analyses d'images de télédétection	agriculture, géologie, météorologie, ressources terrestres, surveillances militaires...	satellites, radar...

Tableau 2 : Domaines d'applications de la Reconnaissance des Formes

1.2 Principe de la reconnaissance de formes

Comme pour tout système intelligent, le principe de la reconnaissance de formes est basé sur la notion d'apprentissage qu'il est nécessaire de bien effectuer pour obtenir de bons résultats. Il s'agit donc d'extraire les paramètres caractéristiques des objets à reconnaître. A ces paramètres, on applique l'opération de normalisation, de réduction et de sélection. Pour former les classes des différents objets en présence, la classification automatique est utilisée. La reconnaissance ou la décision consiste donc à affecter l'objet à reconnaître à la classe la plus appropriée.

1.3 Normalisation et Réduction des paramètres

1.3.1 Normalisation des paramètres

Sachant que les paramètres sont de natures différentes, les grandeurs de leurs mesures peuvent aussi être différentes [3]. Pour éviter ce problème, on fait souvent appel à leur normalisation en utilisant généralement la formule suivante :

$$x_{ij} = \frac{x_{ij} - x_{\min}}{x_{\max} - x_{\min}}, \quad i = 1, 2, \dots, n, \quad j = 1, 2, \dots, m \quad (2)$$

où x_{ij} sont les coordonnées de l'objet O_i et x_{\min} et x_{\max} respectivement les valeurs minimales et maximales de x_{ij} .

Une autre manière de normaliser ces paramètres consiste à centrer, d'abord, le tableau objet/paramètre et le réduire, ensuite, en divisant les composantes x_{ij} de chaque paramètre p_i par son écart type. On aura :

$$x_{ij} = \frac{x_{ij} - \frac{1}{n} \sum_{i=1}^n x_{ij}}{\sigma_j} \quad \text{avec:} \quad \sigma_j^2 = \frac{1}{n} \sum_{i=1}^n \left[x_{ij} - \frac{1}{n} \sum_{i=1}^n x_{ij} \right]^2 \quad (3)$$

Signalons toutefois que la normalisation peut parfois présenter des effets négatifs sur la présentation des données [4].

1.3.2 Réduction des paramètres

Le choix des paramètres ne devra pas être effectué d'une manière subjective. Au début, il est conseillé de prendre en compte tout paramètre mesurable et calculable. L'élimination des paramètres n'apportant pas beaucoup d'information doit se faire automatiquement en appliquant, par exemple, l'Analyse en Composantes Principales (ACP) [5]. Rappelons que cette méthode est essentiellement descriptive. Son objectif est de présenter, sous forme graphique, le maximum de l'information et son

principe consiste à remplacer une variable aléatoire multidimensionnelle de dimension m par une variable de dimension inférieure en éliminant la redondance d'information qui est due aux liaisons existant entre plusieurs variables. Les droites et les plans issus d'une ACP ne sont pas réalisés avec les variables initiales mais avec des indices obtenus par leurs combinaisons linéaires. Parmi tous les indices possibles, l'A.C.P. recherche d'abord celui qui permet de voir au mieux les individus et pour lequel la variance des individus est maximum. Cet indice est appelé première composante principale ou encore premier axe principal. Ensuite, un deuxième indice est recherché et ceci sous deux conditions, à savoir, avoir une corrélation linéaire nulle avec la première et avoir, à son tour, la plus grande variance. Ainsi, cette deuxième composante principale fournit la plus grande information possible, complémentaire à la première. Ce processus se déroule jusqu'à l'obtention de la dernière composante principale. Sur un tableau de données où n est le nombre d'objets ou individus et m , le nombre de paramètres prélevés sur chacun de ces objets, la méthode s'applique en suivant les opérations suivantes :

- calcul des moyennes des vecteurs colonnes,
- centrage des variables,
- calcul du vecteur des écarts-types,
- calcul de la matrice de données centrées réduites,
- calcul de la matrice des corrélations si l'on veut donner le même poids à toutes les variables sinon de la matrice de variance-covariance. A la matrice des corrélations, on fait subir une opération mathématique qui va fournir la matrice diagonale des valeurs propres et la matrice des vecteurs propres,
- calcul de la matrice des nouvelles coordonnées des individus par rapport aux axes principaux. Les dimensions de cette dernière matrice

sont réduites par rapport aux précédentes et les individus sont représentés sur des espaces de dimensions réduites, à savoir, le premier plan principal formé par les deux premiers axes principaux et les autres plans principaux formés avec les autres axes significatifs.

1.4 Classification Automatique

Après la réduction des paramètres, on applique la classification automatique pour former les classes en présence [6,7,8,9]. Parmi les méthodes qui existent, les plus utilisées sont de type statistique. En fonction de l'information *a priori*, ces méthodes sont supervisées ou non supervisées.

1.4.1 Décision bayésienne

Dans le cas où nous aurions toutes les informations en termes de probabilités *a priori* sur les objets, nous pouvons utiliser le théorème de Bayes qui permet de transformer ces probabilités *a priori* en probabilités *a posteriori*, c'est-à-dire à trouver la classe la plus probable [10]. Il s'agit de déterminer la meilleure classe $w_i(x)$ ou la classe la plus probable en appliquant la formule suivante :

$$\omega^*(x) = w_j / \forall i, 1 \leq i \leq K, p(x/w_j)p(w_j) \geq p(x/w_i)p(w_i) \quad (4)$$

ou, autrement écrite :

$$p(w_i/x) = \frac{p(x/w_i)p(w_i)}{\sum_{i=1}^K p(x/w_i)p(w_i)} \quad (5)$$

Dans le cas où l'information n'est pas complète, on fait appel aux méthodes supervisées (ou avec professeur) et non supervisées (ou sans professeur). On utilisera les premières dans le cas où les données sont classées *a priori* et on forme soi-même les classes. On utilisera les

secondes dans le cas où l'on ne dispose pas d'information *a priori*. Les recherches actuelles s'orientent de plus en plus vers les méthodes non paramétriques non supervisées ou les méthodes de recherche de groupements ou "clusterings" [11,12,13].

1.4.2. Découpage en fenêtres de Parzen

Le but des fenêtres de Parzen est de donner une estimation de $p(X/w_i)$ pour la classe w_i [14]. Dans le cas où le domaine D est par exemple un hypercube à m dimensions et d'arête ayant pour longueur h .

Son volume est : $V = h^m$ (6)

Cet estimateur est calculé par la formule suivante :

$$p(X) = \frac{1}{N} \sum_{i=1}^N \frac{1}{V} \Phi\left(\frac{X_0 - X_i}{h}\right) \quad (7)$$

où la fonction $\Phi\left(\frac{X_0 - X_i}{h}\right)$ prend la valeur 1 si un échantillon se trouve à l'intérieur de l'hypercube et prend la valeur nulle dans le cas contraire. Le nombre de prototypes à l'intérieur de l'hypercube est donné par :

$$k = \sum_{i=1}^N \Phi\left(\frac{X_0 - X_i}{h}\right) \quad (8)$$

Les objets ayant le même estimateur peuvent être regroupés dans une même classe. Cette méthode fait partie des méthodes de classification automatiques non paramétriques et supervisées.

1.4.3. Classification par les k plus proches voisins

Contrairement à la méthode précédente, on augmente le volume d'un hypercube jusqu'à ce qu'il renferme le nombre fixé k_D d'objets pour

ensuite déterminer l'estimateur de chaque objet donné par la formule suivante [15] :

$$\hat{P}(x_i) = \hat{P}(x_i / x_i \in D) = \frac{k_D}{V(D)} \quad (9)$$

où $V(D)$ est le volume de l'hypersphère centrée en x_i et refermant k_D observations. Ce nombre k_D est souvent la racine carrée du nombre total N des observations. Pour le calcul du volume $V(D)$, on utilise l'expression (8) où d est la distance entre x_i et son k -ième plus proche voisin:

$$V(D) = \alpha e^{k_D \text{Log } d} \quad (10)$$

$$\text{avec : } \alpha = \frac{e^{k_D \text{Log } \pi}}{\Gamma(k_n)}$$

où $\Gamma(\cdot)$ est la fonction GAMMA. Les observations dont les estimateurs sont semblables forment une même classe. Notons que cette méthode est attrayante par sa simplicité d'utilisation car elle fournit des résultats intéressants. La possibilité de projeter les données suivant deux axes choisis par l'utilisateur peut fournir des informations quant à la caractéristique discriminante d'un paramètre [16].

1.4.4 Méthodes hiérarchiques

Les méthodes hiérarchiques peuvent être ascendantes ou descendantes, la procédure suit un arbre hiérarchique. Le niveau des nœuds indique le degré de ressemblance entre les objets correspondants. En coupant l'arbre à différents niveaux, on obtient des partitions. Ces méthodes sont basées sur la fusion d'objets décidée par un critère utilisant la dispersion intraclasse et en utilisant le théorème de Huyghens [17]. Notons toutefois que la fusion de deux classes s'effectue non sur un calcul de distances entre classes mais sur l'augmentation de la dispersion intraclasse. Ces méthodes consistent à diviser d'abord l'ensemble des

données par dichotomie en deux classes, puis chacune de ces classes est elle-même divisée et ainsi de suite jusqu'à ce que les subdivisions soient homogènes ou d'effectif faible pour se prêter à une partition sûre. La classe à fractionner est celle pour laquelle l'augmentation du moment intraclasse est maximum. Si les méthodes hiérarchiques donnent, en résultat, une série de partitions emboîtées qu'il suffit d'interpréter par l'arbre hiérarchique associé, celles-ci sont des méthodes lentes et ne fournissent des résultats satisfaisants que dans le cas où les données sont bien distinctes.

1.4.5 Méthode d'agrégation autour des centres mobiles

Le principe de la méthode d'agrégation autour des centres mobiles ou nuées dynamiques est de fixer une partition initiale puis de recalculer les centres de gravité de ces partitions à chaque fois que l'on réaffecte les individus à la classe ayant comme centre de gravité le plus proche de cet objet [18]. L'opération de réaffectation est répétée jusqu'à ce que l'on ait l'état stationnaire. Ce dernier est caractérisé par la stabilité des individus, point de convergence de la méthode. Cette convergence est assurée si la partition initiale du nuage de points est bien choisie. Le moment centré d'ordre deux utilise un calcul de distance euclidienne.

1.4.6 Méthode mixte

Dans la méthode proposée [19], les données multidimensionnelles sont visualisées dans un espace bidimensionnel. L'utilisateur fait déplacer un observateur fictif $H(h_1, h_2, \dots, h_m)$ et fixe la direction de "son regard" $E(\mu_1, \mu_2, \dots, \mu_m)$ désignée par un vecteur de référence appelé "vecteur regard" (fig.1).

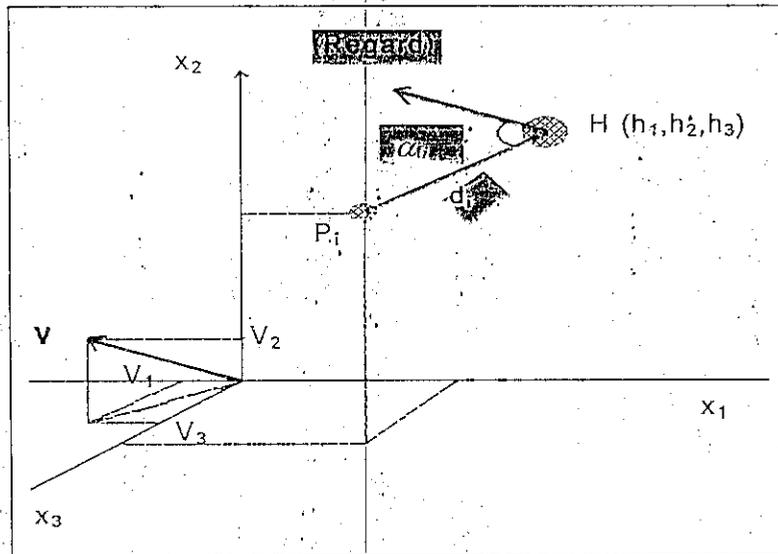


Fig.1 : Schéma de principe de la méthode mixte

Toutes les données multidimensionnelles sont alors repérées :

- d'une part, par les distances d_i qui les séparent de l'observateur
- et, d'autre part, par leurs positions angulaires α_i par rapport au vecteur regard comme le montrent les équations suivantes :

$$d_i = \sqrt{\sum_{j=1}^m (x_{ij} - h_j)^2}, \quad i=1,2,\dots,n \quad (11)$$

$$\alpha_i = \cos^{-1} \frac{\sum_{j=1}^m (x_{ij} - h_j) v_j}{d_i \sqrt{\sum_{j=1}^m v_j^2}}, \quad i=1,2,\dots,n \quad (12)$$

où :

- x_{ij} sont les coordonnées de l'objet i ,
- h_j les coordonnées de l'observateur
- v_j les projections du vecteur regard sur les axes de l'espace de représentation.

Dans le nouveau repère bidimensionnel dont les deux coordonnées sont constituées par la distance à l'observateur et l'angle par rapport au vecteur regard, les données sont visualisées graphiquement dans le contexte de leurs regroupements en classes. Le choix d'autres positions de l'observateur ou d'autres directions du vecteur regard entraîne différentes visualisations graphiques ou différents regroupements. Parmi les différentes visualisations, l'utilisateur choisit celle qui lui permet de distinguer au mieux les classes formées. Si les classes ne se chevauchent pas, l'utilisateur peut séparer lui-même ces classes en les entourant à l'aide de la souris pour les valider. Dans le cas où il ne peut pas prendre de décision lorsque les classes se chevauchent, il fixe les centres des classes et fait appel à la méthode d'Agrégation Autour des Centres Mobiles, dont les performances sont justement améliorées grâce au choix précis des centres des partitions initiales. Cette manière de combiner les deux parties, la partie visualisation graphique et la partie automatique de la classification, offre l'avantage de faire intervenir l'une dans le cas où l'autre perd de son efficacité.

1.5 Conclusion

Indispensable pour l'automatisation complète de certaines tâches industrielles comme le tri, le contrôle de qualité, la surveillance, etc., la reconnaissance des formes intéresse de plus en plus de nombreux chercheurs : automaticiens, informaticiens, mathématiciens. Quelle que soit l'utilisation, le principe de base est le même puisqu'on retrouve les phases de paramétrisation, d'apprentissage et de reconnaissance ou de décision. Au cours de cette phase d'apprentissage ou de classification automatique, on peut se retrouver devant différentes situations selon la disposition ou non de l'information *a priori*. Les méthodes de

classification automatique sont nombreuses et diversifiées. Dans ce chapitre, nous avons rappelé les plus importantes alors qu'il existe d'autres telles les surfaces séparatrices, la classification séquentielle, la classification avec rejet, la classification par regroupement etc.

Dans le cas où le problème de reconnaissance consiste à trier des objets, avant d'extraire l'ensemble des paramètres nécessaires à la classification, nous devons faire subir à l'image de l'objet un certain nombre de traitements préliminaires comme nous allons le voir dans le prochain chapitre.

Chapitre 2

Traitement d'Images en Reconnaissance de Formes

2.1 Introduction

En reconnaissance des formes par vision artificielle, avant l'extraction des paramètres caractéristiques de l'objet, quelques opérations de traitement d'images sont nécessaires. Il s'agit essentiellement d'opération de lissage, de filtrage, de binarisation, de suivi de contour etc. Par ailleurs, les paramètres calculés peuvent être morphologiques, de couleur ou de texture.

2.2 Acquisition des images des graines

Les images des graines acquises à l'aide d'une caméra vidéo et d'une carte d'acquisition sont enregistrées en niveaux de gris et en fausses couleurs sous format BMP, TIF, GIF, JPEG, MPEG etc. (fig.1). En couleur, ces images sont codées sur vingt quatre bits dans trois plans mémoire où chaque plan correspond aux valeurs des couleurs de base, le Rouge, le Vert et le Bleu.

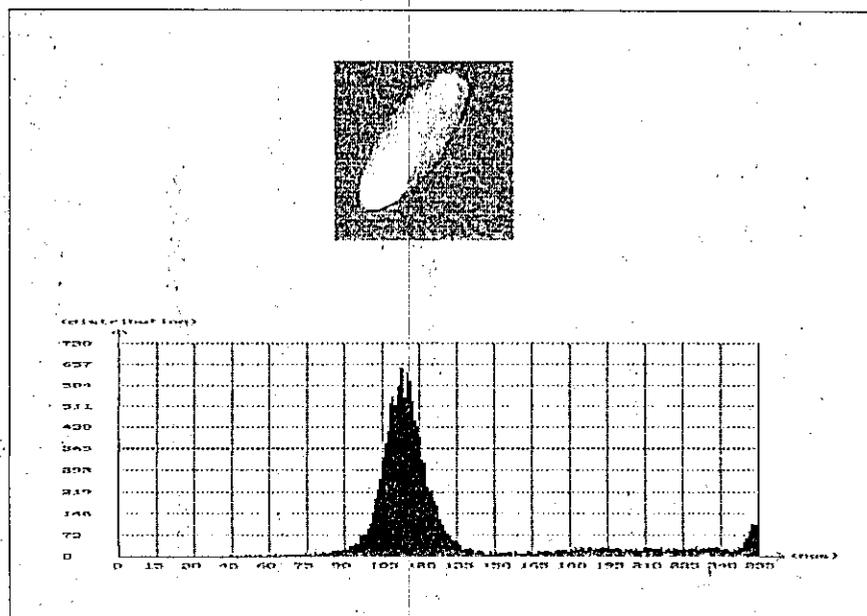


Fig.1 : Image d'une graine en niveaux de gris et son histogramme

2.3 Images en couleur (fig.2).

La nature de la lumière et de la couleur était peu appréhendée avant l'époque de Newton qui a prouvé l'aspect spectral de la lumière par l'expérimentation méticuleuse de la lumière solaire à travers un prisme. L'aspect physique de la couleur qui reste la base de la colorimétrie a été ainsi établi. Par ailleurs, la prévalence de la vision de l'homme a contribué à la confusion entre la vision des couleurs et la nature de la lumière expliquée grâce à la théorie des ondes électromagnétiques. Plusieurs auteurs ont travaillé dans ce domaine [20]. Cependant, c'est Grassman [21] qui a étudié les lois expérimentales du mélange des couleurs qui portent actuellement son nom. De son côté, Maxwell [22] démontre que n'importe quelle couleur peut être mélangée avec des proportions déterminées des trois couleurs primaires. Dans la même époque Helmholtz [23] explique la distinction entre la synthèse soustractive, la synthèse additive et la trichromie en terme de courbes de sensation spectrale des trois cellules sensibles à la couleur dans l'œil. Il a été établi que n'importe quelle couleur peut être obtenue par le mélange additif de trois couleurs primaires qui peuvent ne pas être nécessairement présentes toutes les trois. Les tableaux de la Commission Internationale de l'Eclairage (CIE) contiennent plus de 1000 couleurs par rapport aux sept composantes de l'arc-en-ciel. Ainsi, une reconnaissance d'objets peut se faire à partir des paramètres de leurs couleurs, à savoir, le contraste, la luminance, la saturation et la teinte.

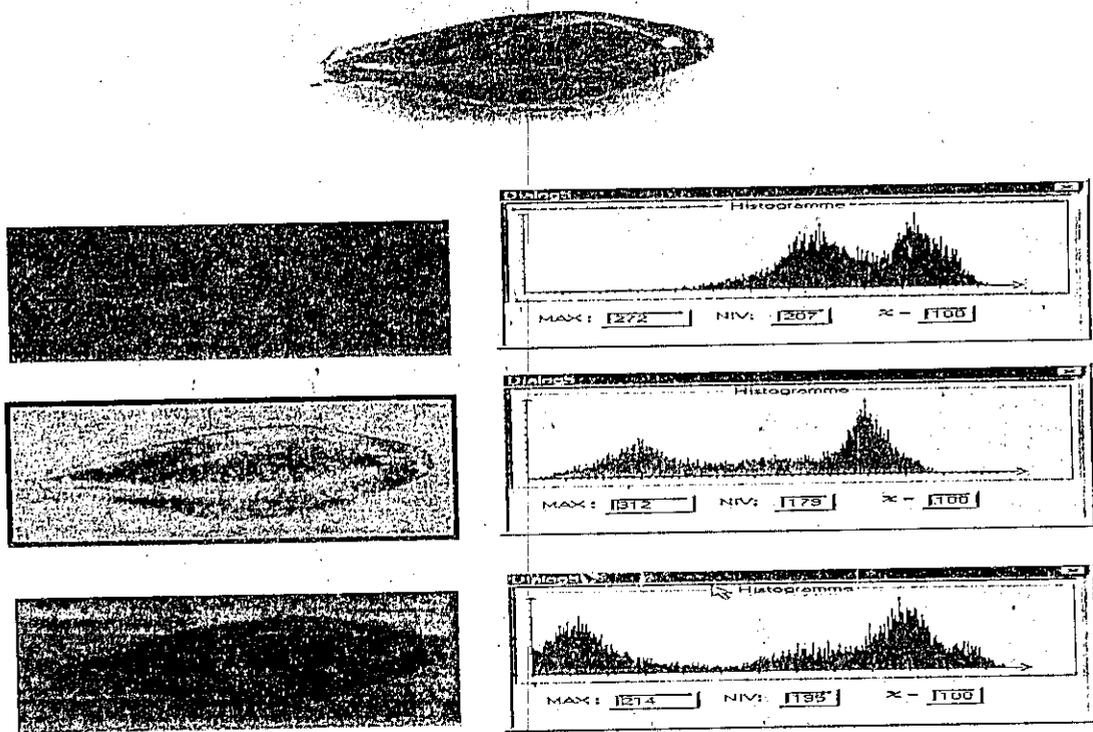


Fig.2 : Images en couleur d'une graine et ses composantes RVB

2.3.1 Système de représentation de la couleur

La différenciation et la compréhension de l'organisation des couleurs peuvent être effectuées par différentes représentations [24]. La première est le cercle chromatique sur lequel sont disposées à équidistance les trois couleurs primaires. Diamétralement opposées à celles-ci, sont disposées respectivement le cyan, le magenta et le jaune. Pour définir les couleurs de manière plus rationnelle par des chiffres, on les quantifie dans les systèmes de référence RVB et CMJ. Ainsi, une couleur quelconque C peut être définie par un triplet de coordonnées trichromatiques de la couleur C dans le système RVB tel que $C = rR + vV + bB$ où les trois mesures r, v et b sont les quantités de chacune des couleurs Rouge, Verte et Bleue. Toutes les couleurs sont à

l'intérieur d'un cube dont les axes sont R, V et B. C'est le type de repère largement utilisé, particulièrement en télévision. Cependant, ce type de repère ne prend pas en charge l'aspect perception des couleurs. C'est ainsi que le repère ITS (Intensité, Teinte, Saturation) a été défini par Munsell. Il existe plusieurs manières de transformer mathématiquement l'espace RVB en l'espace ITS. On peut choisir le système de représentation PAL XYZ de la CIE (Commission Internationale de l'Eclairage) qui calcule X, Y, et Z nouvelles couleurs primaires à partir de R, V et B (système PAL) en faisant :

$$X = 2.769R + 1.7518V + 1.130B \quad (1)$$

$$Y = 1.000R + 4.5907V + 0.060B \quad (2)$$

$$Z = 0.000R + 0.0565V + 5.594B \quad (3)$$

Ces trois valeurs sont ensuite transformées en d'autres valeurs L^* a^* b^* de la CIE tel que :

$$L^* = 116 (Y/Y_0)^{1/3} - 16 \quad (4)$$

$$a^* = 500[(X/X_0)^{1/3} - (Y/Y_0)^{1/3}] \quad (5)$$

$$b^* = 200[(Y/Y_0)^{1/3} - (Z/Z_0)^{1/3}] \quad (6)$$

où X_0 , Y_0 et Z_0 sont les valeurs correspondantes au blanc de référence.

2.3.2. Paramètres de couleur

Les paramètres de couleur habituellement utilisés sont la teinte, l'intensité lumineuse et la saturation (I T S) et sont respectivement calculés par :

$$I = L^* \quad (7)$$

$$T = \text{Arctang}(b^* / a^*) \quad (8)$$

$$S = \sqrt{(a^*)^2 + (b^*)^2} \quad (9)$$

La signification des ces trois importants paramètres est la suivante :

2.3.2.1. L' intensité

L'intensité constitue le degré d'éclaircissement ou d'assombrissement des couleurs. La luminance d'une couleur correspond à la quantité d'énergie réfléchié par l'objet coloré. Dans le langage courant, la luminance correspond aux termes suivants : vif, lumineux, clair, sombre. Réglée au minimum, celle-ci est si faible que l'image disparaît de l'écran (écran «noir»). Réglée au maximum, la luminosité de l'écran est telle que l'image disparaît également (écran blanc). Ce réglage est souvent appelé «luminosité» ou «lumière».

2.3.2.2. La teinte

La teinte représente la qualité spécifique de sensation ou bien la qualité dominante de la couleur qui est due à une radiation de longueur d'onde déterminée. Le Gris ne correspond à aucune teinte tout comme le Noir ou le Blanc qui n'en sont que des cas particuliers. Dans ce cas, la teinte est indéterminée ou quelconque.

2.3.2.3 La saturation

La saturation est inversement proportionnelle à la radiation blanche qui affaiblit la coloration du rayonnement. En effet, les couleurs des objets sont en général toujours plus ou moins diluées de blanc. On retrouve cette notion, par exemple, en peinture où l'on ajoute du blanc à une couleur de façon à diminuer sa saturation comme par exemple le rose qui est un rouge désaturé ou pâle. On notera que les couleurs spectrales (monochromatiques) sont très pures (très saturées). Par contre, un gris a une saturation nulle. A partir des

images couleur codées sur 24 bits sous le format BMP, les trois composantes R, V et B sont extraites et les paramètres de couleur ITS peuvent être calculés.

2.4. Images texturées

Aujourd'hui, on s'oriente de plus en plus vers l'étude d'images de texture vue la richesse de l'information qu'elle véhicule. En effet, la texture qui est une propriété inhérente aux surfaces, caractérise la granularité, la répétitivité, la directivité et elle n'est pas inhérente au niveau de gris lui-même mais aux relations spatiales qui le lient à ses voisins [25,26]. Les paramètres qui caractérisent la texture sont nombreux et différent en fonction de la méthode utilisée. Parmi ces méthodes, on peut citer la méthode :

- de la dépendance spatiale des niveaux de gris,
- de la densité spectrale,
- des longueurs d'iso-segments de niveau de gris,
- des différences de niveau de gris.

Dans la *Méthode de la Densité Spectrale* (PSM : Power Spectral Method), le spectre de puissance d'une image $x(k,l)$ est calculé en utilisant la transformée de Fourier de cette image. A partir du spectre de puissance de l'image, plusieurs paramètres sont calculés. Dans la *Méthode de Différence de Niveaux de Gris* (GLDM : Gray Level Différence Method), des paramètres de texture sont calculés à partir de la matrice de différence de niveaux de gris entre les pixels de l'image. Dans la *Méthode des Longueurs d'Iso-segments de Niveaux de Gris* (GLRLM : Gray Level Run Length Method), on détecte les suites colinéaires de points ayant le même niveau de gris dans une direction donnée. La longueur d'un iso-segment est le nombre de points contenus dans l'iso-segment. A partir

de la matrice (nombre d'iso-segments, longueur des iso-segments), on peut calculer cinq paramètres de texture. Parmi ces méthodes, la plus employée est certainement la *Méthode de la Dépendance Spatiale des Niveaux de Gris* (SGLDM : Spatial Gray Level Dépendance Method) qui est basée sur la notion de matrice de cooccurrence des niveaux de gris.

Si $I(x, y)$ est la fonction image définie sur un domaine D à N_g niveaux de gris de sorte que $I(x, y) = 0, 1, 2, \dots, N_g$ et si R est une relation spatiale liant deux pixels, la matrice de cooccurrence relative à la relation R est définie par l'expression (10) ci-dessous où $M(i, j)$ est le nombre de couples de points du domaine D liés par la relation R , le premier ayant le niveau de gris i , et l'autre le niveau de gris j [27].

$$M(i, j) = \{ \{ (x, y), (x', y') \} \in D, \text{ vérifiant } R / I(x, y) = i, I(x', y') = j \} \quad \begin{matrix} i = 0, 1, \dots, N_g - 1 \\ j = 0, 1, \dots, N_g - 1 \end{matrix} \quad (10)$$

Divisé par le nombre total de couples, $M(i, j)$ devient une probabilité telle que :

$$m(i, j) = \frac{M(i, j)}{\text{Card} \{ \{ (x, y), (x', y') \} \in D \text{ vérifiant } R \}} \quad (11)$$

Les relations R sont définies de plusieurs manières. Dans notre cas, R désigne la distance d entre deux pixels selon l'orientation θ de la droite joignant ces deux pixels par rapport à l'horizontale. Les matrices de cooccurrence dépendent de la distance d et de θ . Dans la pratique, la distance d est égale à 1 et la direction θ prend généralement les quatre valeurs $\pi/4$, $\pi/2$, $3\pi/4$ et 2π par rapport à l'horizontale. De la matrice de cooccurrence dont la taille est fonction du nombre N_g de niveaux de gris de l'image, quatre paramètres texturaux sont calculés de la manière suivante :

- le *Moment Angulaire du Second ordre* (MAS) qui constitue une mesure de l'hétérogénéité de la texture et dont la valeur augmente avec l'homogénéité de la texture:

$$SMA = \sum_{i=0}^{N_x-1} \sum_{j=0}^{N_x-1} (m(i,j))^2 \quad (12)$$

- la *Corrélation* qui quantifie la dépendance directionnelle des niveaux de gris et qui atteint ses plus grandes valeurs lorsque la direction θ est voisine de l'orientation des lignes de texture :

$$COR = \frac{\sum_{i=0}^{N_x-1} \sum_{j=0}^{N_x-1} (i - \mu_x)(j - \mu_y)m(i,j)}{\sigma_x \sigma_y} \quad (13)$$

avec:

$$\mu_x = \sum_{i=0}^{N_x-1} i \sum_{j=0}^{N_x-1} m(i,j)$$

$$\mu_y = \sum_{i=0}^{N_x-1} j \sum_{j=0}^{N_x-1} m(i,j)$$

$$\sigma_x^2 = \sum_{i=0}^{N_x-1} (i - \mu_x)^2 \sum_{j=0}^{N_x-1} m(i,j)$$

$$\sigma_y^2 = \sum_{i=0}^{N_x-1} (i - \mu_y)^2 \sum_{j=0}^{N_x-1} m(i,j)$$

où μ_x et μ_y sont des déviations standard.

- l'*Entropie* qui permet de caractériser le degré d'organisation ou de désordre dans l'image et est minimale pour une texture homogène.

$$ENT' = - \sum_{i=0}^{N_x-1} \sum_{j=0}^{N_x-1} m(i,j) \log m(i,j) \varepsilon \quad (14)$$

avec $\varepsilon=1$ si $m(i,j) \neq 0$

$\varepsilon=0$ sinon

- le *moment inverse* dont la valeur devient de plus en plus faible, lorsque l'on s'éloigne de la diagonale principale. Il est proche de l'unité s'il y a une concentration de points autour de cette diagonale. Ce paramètre permet de mettre en évidence les plages uniformes de texture:

$$MI = \frac{\sum_{i=0}^{N_x-1} \sum_{j=0}^{N_y-1} m(i, j)}{1 + (i + j)^2} \quad (15)$$

2.5 Filtrage

Une image en niveaux de gris à l'état brut ne permet pas une extraction directe des informations caractérisant l'objet qu'elle contient si les conditions de son acquisition (éclairage, qualité du capteur, etc.) offrent des images bruitées. Dans ce cas, l'utilisation du filtrage s'avère indispensable avant tout autre traitement [28,29]. Rappelons que les trois familles principales de filtres sont linéaires, non linéaires et morphologiques. Un filtre est dit linéaire s'il affecte à chaque point $g(i, j)$ de l'image une combinaison linéaire de ses voisins. Parmi les filtres linéaires, on peut citer le filtre passe-bas qui consiste à remplacer la valeur de chaque pixel $g(i, j)$ par le produit de convolution de la matrice de voisinage suivante :

$g(i-1, j-1)$	$g(i-1, j)$	$g(i-1, j+1)$
$g(i, j-1)$	$g(i, j)$	$g(i, j+1)$
$g(i+1, j-1)$	$g(i+1, j)$	$g(i+1, j+1)$

et le masque suivant :

$$H = \begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline 2 & 4 & 2 \\ \hline 1 & 2 & 1 \\ \hline \end{array}$$

Ce type de filtrage atténue les variations de lumière. Ils ont pour effet d'adoucir les contours de l'image et d'éliminer le bruit du fond de cette image. Quant au filtre passe-haut, il est conçu pour augmenter le contraste de l'image et permet la mise en évidence de contours entre les plages de niveaux de gris différents. Le masque le plus utilisé est :

$$H = \begin{array}{|c|c|c|} \hline -3 & 1 & -3 \\ \hline 1 & 8 & 1 \\ \hline -3 & 1 & -3 \\ \hline \end{array}$$

Parmi les filtres linéaires utilisés, on peut encore mentionner le filtre moyen qui consiste en le remplacement du niveau de gris du point $g(i,j)$ par la moyenne des niveaux de gris de ses voisins. Cette opération est appelée aussi lissage et provoque un flou sur l'image. Le masque correspondant est :

$$H = \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$

Parmi les filtres non linéaires, on peut citer le filtre médian et les filtres morphologiques. Le principe de ces filtres est de remplacer la valeur du

niveau de gris du point $g(i,j)$ par la valeur médiane de ses voisins. Comme exemple, prenons la fenêtre suivante dont le pixel central a pour valeur 13.

$$H = \begin{array}{|c|c|c|} \hline 28 & 7 & 200 \\ \hline 1 & 13 & 129 \\ \hline 0 & 40 & 100 \\ \hline \end{array}$$

Après l'application du filtre médian, ce pixel central prend la valeur 28 puisque la médiane des valeurs contenues dans le fenêtre est 28.

$$H = \begin{array}{|c|c|c|} \hline 28 & 7 & 200 \\ \hline 1 & 28 & 129 \\ \hline 0 & 40 & 100 \\ \hline \end{array}$$

Quant aux filtres morphologiques, ce sont des filtres obéissant aux deux conditions morphologiques d'idempotence ($T^2 f(x) = T f(x)$) et de croissance (si $f(x) \geq g(x)$ alors $T f(x) \geq T g(x)$). Ce genre de filtres se base sur deux principes fondamentaux qui sont l'*Erosion* et la *Dilatation*. On rappelle que l'érosion d'une image consiste à choisir un élément structurant de forme quelconque (carré, rectangle, etc.) et à attribuer à chaque point du domaine B_x la valeur inférieure que peut prendre $f(x)$ dans ce domaine. Son application sur l'image réduit les pics et élargit les vallées. On peut formuler le principe par :

$$E^B f(x) = \text{Inf} \{ f(u) / u \in B_x \} \quad (16)$$

A l'inverse de l'érosion, la dilatation d'une image attribue la valeur supérieure dans la forme choisie (carré, cercle, etc.). Son application épaissit les pics et comble les vallées. On peut formuler le principe par :

$$D^B f(x) = \text{Sup} \{ f(u) / u \in B_x \} \quad (17)$$

On montre que ces deux filtres ne sont pas des filtres morphologiques du moment qu'ils ne vérifient pas les deux conditions citées ci-dessus.

L'ouverture est l'application de l'érosion suivie de sa dilatation. On peut formuler le principe par :

$$O^B f(x) = E^B D^B f(x) = \text{Sup}\{E^B f(y) / y \in B_x\} = \text{Inf}\{\text{Sup}\{f(z) / z \in B_y\}\} \quad (18)$$

Quant à la fermeture, c'est l'application de la dilatation suivie de l'érosion :

$$F^B f(x) = D^B E^B f(x) = \text{Sup}\{D^B f(y) / y \in B_x\} = \text{Sup}\{\text{Inf}\{f(z) / z \in B_y\}\} \quad (19)$$

Comme pour la dilatation et l'érosion on peut combiner l'ouverture et la fermeture. L'application de ces filtres dans cet ordre donne naissance à la technique dite « Ouverture Fermeture Ouverture » ou OFO :

$$f(x) = (E^B D^B)(D^B E^B)(E^B D^B)f(x) = E^B D^{2B} E^{2B} D^B f(x) \quad (20)$$

On peut encore avoir « Fermeture Ouverture Fermeture » ou FOF qui peut s'exprimer comme suit :

$$f(x) = (D^B E^B)(E^B D^B)(D^B E^B)f(x) = D^B E^{2B} D^{2B} E^B f(x) \quad (21)$$

Parmi les filtres proposés, le filtre FOF (Fermeture Ouverture Fermeture) donne de bons résultats dans l'élimination des pics et des vallées ainsi que l'élimination des taches séparées étant donné qu'il se base sur les deux principes de l'érosion et de la dilatation. La figure 3 montre quelques exemples de filtres.

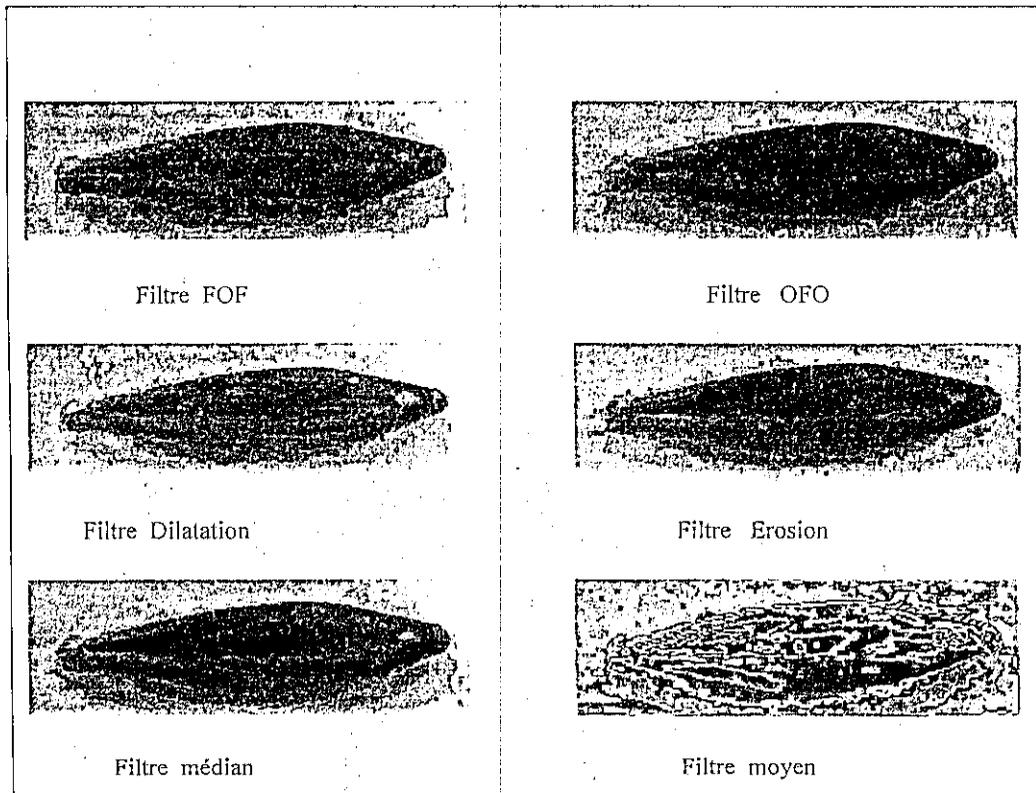


Fig.3 : Exemples de filtres morphologiques

2.6 Binarisation des images

En traitement d'images, si la segmentation consiste à séparer les régions homogènes, la binarisation permet surtout d'extraire les objets de leur fond. En appliquant l'opération de suivi de contours à l'image binarisée, le contour est obtenu d'une manière plus rapide et plus précise que l'application des opérateurs différentiels tels que le Gradient, le Laplacien etc.

Dans ce qui suit, on note $L=\{0,1,2,\dots,l-1\}$ l'ensemble des niveaux de gris, $I(x,y)$ la luminance d'un point de coordonnées (x,y) , t le seuil permettant de binariser une image et t' le seuil optimum par rapport à un critère déterminé. Ainsi, dans sa forme la plus simple, la binarisation consiste, pour une image comportant des

objets clairs sur un fond sombre, à choisir un niveau de gris t et à affecter à chaque pixel le label objet ou fond, conformément au test suivant :

- si $I(x,y) > t$, les pixels sont pris comme étant des objets.
- si $I(x,y) < t$, les pixels sont pris comme étant le fond.

A partir de l'image binarisée, des paramètres morphologiques autres que texturaux sont extraits de chaque objet puis rangés sous formes de tableaux multidimensionnels. A ce tableau de données, on applique la classification automatique pour former les classes d'objets homogènes auxquelles seront affectés les objets inconnus lors de la reconnaissance. Plusieurs méthodes de binarisation d'images ont été proposées à tel point que l'utilisateur non averti reste souvent indécis quant au choix de la méthode à utiliser. Dans notre travail, sept principales méthodes de binarisation basées sur un seuillage global ou local sont examinées pour mettre en évidence les avantages et les inconvénients de chacune d'elles sur la base de mesures sur l'uniformité, le contour et le coût informatique.

2.6.1. Méthodes de binarisation par seuillage global

Parmi les méthodes de binarisation par seuillage global, on peut distinguer celles qui déterminent le seuil à partir de l'ensemble des niveaux de gris de l'image tout entière de celles dont le seuil est calculé en tenant compte des propriétés locales de chaque point. Parmi les méthodes globales de binarisation, celles auxquelles nous sommes particulièrement intéressés sont les méthodes utilisant la maximisation d'entropie, la maximisation de la variance interclasse, la minimisation de l'erreur quadratique et la conservation des moments.

2.6.1.1. *Seuillage par maximisation de l'entropie*

Les méthodes de binarisation utilisant la notion d'entropie consistent à maximiser la bande limite de l'entropie *a posteriori* afin de rendre maximum les

contrastes entre les régions. Parmi ces méthodes, on peut citer la méthode de Pun [30], la méthode de Kapur [31] et celle de Johansen et Billie [32]. Connaissant l'entropie de l'histogramme des niveaux de gris, Pun a proposé un algorithme où le seuil optimal t^* est calculé par:

$$\sum_{i=0}^{t^*} p_i = \begin{cases} -\alpha & \text{si } \alpha \leq 0.5 \\ +\alpha & \text{si } \alpha > 0.5 \end{cases} \quad (22)$$

où α est le coefficient d'anisotropie tel que:

$$\alpha = \frac{\sum_{i=0}^m p_i \text{Log}_e(p_i)}{\sum_{i=0}^{l-1} p_i \text{Log}_e(p_i)} \quad (23)$$

Dans cette expression m est le plus petit entier tel que:

$$\sum_{i=0}^m p_i \geq 0.5 \quad (24)$$

Cependant cette méthode qui donne un seuil optimal $t^* > m$ introduit du biais et, dans le cas d'images bruitées, les contours sont disjoints et la segmentation excessive. Pour remédier à ce problème, dans sa méthode, Kapur définit le seuil optimum t^* comme étant le niveau de gris qui maximise $H_b(t) + H_n(t)$. Ce seuil est donné par:

$$t^* = \text{ArgMax}_{t \in G} (H_b(t) + H_n(t)) \quad (25)$$

où :

$$H_b = \sum_{i=0}^t p_i \text{Log}(p_i) \quad (26)$$

$$H_n = \sum_{i=t+1}^{l-1} p_i \text{Log}(p_i) \quad (27)$$

2.6.1.2 Seuillage par maximisation de la variance interclasse

Dans cette méthode basée sur la classification automatique, il s'agit de séparer les pixels d'une image en deux classes C_1 et C_2 qui correspondent

respectivement à l'objet et au fond [33]. Le seuil optimum est déterminé en maximisant la variance interclasse donnée par l'expression suivante :

$$\sum_{j=1}^2 P(C_j)(M_j - M)^2 \quad (28)$$

où :

- $P(C_j)$ est la probabilité qu'un pixel appartienne à la classe C_j
- M_j la moyenne des niveaux de gris de cette classe.
- M est le niveau de gris moyen de l'image.

Le seuil obtenu assure une bonne détection des contours pour une grande variété d'images.

2.6.1.3 Seuillage par minimisation de l'erreur quadratique

En considérant l'histogramme des niveaux de gris comme une estimée de la fonction de densité de probabilité $p(n)$ d'un mélange de population formée des niveaux de gris des objets et du fond, on peut supposer que chacune des deux composantes $p(n/i)$ du mélange est une distribution normale avec une moyenne μ_i , un écart type σ_i et une probabilité *a priori* P_i . Le seuil optimal est obtenu en minimisant la fonction $J(t)$ suivante [34] :

$$J(t) = 1 + 2 [P_1(t) \text{Log } \sigma_1(t) + P_2(t) \text{Log } \sigma_2(t)] - 2 [P_1(t) \text{Log } P_1(t) + P_2(t) \text{Log } P_2(t)] \quad (29)$$

où :

$$\sigma_1^2(t) = \frac{\sum_{n=0}^t (n - \mu_1(t))^2 h(n)}{P_1(t)}$$

$$\sigma_2^2(t) = \frac{\sum_{n=t+1}^{l-1} (n - \mu_2(t))^2 h(n)}{P_2(t)}$$

avec :

$$P_1(t) = \sum_{n=0}^t h(n) \quad P_2(t) = \sum_{n=t+1}^{l-1} h(n) \quad \text{et} \quad \mu_1(t) = \frac{\sum_{n=0}^t nh(n)}{P_1(t)} \quad \mu_2(t) = \frac{\sum_{n=t+1}^{l-1} nh(n)}{P_2(t)}$$

Le seuil optimal est défini par $t^* = \text{Arg Min} J(t)$, $t \in N_g$. Le seuil obtenu donne des contours qui ne sont pas très précis lorsque les distributions de niveaux de gris des deux classes sont légèrement mélangées.

2.6.1.4 Seuillage par conservation des moments

Dans la méthode du seuillage par conservation des moments, on calcule les seuils en partant du fait que les trois moments de l'image sont préservés entre l'image d'origine et l'image binarisée [35]. Le moment d'ordre i est donné par :

$$m_i = \frac{1}{N} \sum_{n=0}^{l-1} n^i h(n) \quad i = 1, 2, 3, \dots \quad (30)$$

où N est le nombre de pixels de l'image. Le seuil optimal t^* est obtenu à partir de l'histogramme des niveaux de gris correspondant à P_0 % tel que :

$$P_0 = \frac{z - m_1}{(C_1^2 - 4C_0)^{1/2}} \quad (31)$$

où :

$$c_0 = \frac{m_1 m_3 - m_2^2}{m_2 - m_1^2}$$

$$c_1 = \frac{m_1 m_2 - m_3^2}{m_2 - C_1^2}$$

$$z = \frac{1}{2} [(C_1^2 - 4C_0)^{1/2} - C_1]$$

2.6.2. Méthodes de binarisation par seuillage local

Au lieu d'un seuillage global sur toute l'image, celle-ci est divisée en plusieurs sous-images ou fenêtres. Ensuite, on cherche le seuil optimal correspondant à chacune d'elles sur la base de deux méthodes dépendant du pixel ou bien de la région. L'ensemble des méthodes globales précédentes

peuvent être appliquées à chacune de ces fenêtres pour calculer le seuil optimum. Ainsi, parmi les méthodes de seuillage locales proposées, nous nous sommes particulièrement intéressés à deux : celle qui est basée sur la détection de seuils locaux avec test de validation et celle maximisant un critère d'homogénéité.

2.6.2.1 Seuillage local avec test de validation

Cette méthode du seuillage local avec test de validation est basée sur la détection automatique des différents seuils locaux en tenant compte des priorités locales de chaque histogramme, qu'il soit unimodal, bimodal ou multimodal [36]. L'image à traiter de taille $N \times N$ et à I niveaux de gris est subdivisée en plusieurs fenêtres W_j de dimensions $n \times n$. En utilisant deux critères C_1 et C_2 basés sur la mesure des fluctuations des niveaux de gris $I_g(x,y)$, la première opération consiste à mettre en évidence les classes les plus importantes en faisant subir des modifications aux histogrammes correspondant à chaque fenêtre. En fonction de ces deux critères, la fréquence d'apparition $h(i)$ du niveau de gris $I_g(x,y)$ est modifiée de la manière suivante:

$$\begin{aligned} h(i) &= h(i) + 1 && \text{si } C_1 \text{ ou } C_2 \text{ vrai} \\ h(i) &= h(i) + 2 && \text{si } C_1 \text{ et } C_2 \text{ vrais} \\ h(i) &= h(i) && \text{sinon} \end{aligned} \quad (32)$$

où les critères C_1 et C_2 sont définis tels que:

$$C_1 = |I_g(x, y) - I_g(x - 1, y)| \leq 1$$

$$C_2 = |I_g(x, y) - I_g(x, y - 1)| \leq 1$$

La seconde opération consiste à lisser l'histogramme modifié pour éliminer les irrégularités qu'il contient et à remplacer chaque fréquence d'apparition $h(i)$ par la valeur moyenne des fréquences des points voisins.

La troisième opération cherche les seuils optimaux en détectant les extrema principaux locaux contenant les niveaux de gris de l'histogramme lissé dont les fréquences d'apparition sont inférieures à la moyenne des p_i .

Dans le cas où l'histogramme est unimodal, un zéro est affecté à la valeur du seuil t^* . Dans le cas où il est bimodal, on attribue à ce seuil la valeur $(l_{g1} + l_{g2})/2$ où l_{g1} et l_{g2} correspondent respectivement aux niveaux de gris des maxima p_1 et p_2 .

Dans le cas où l'histogramme est multimodal, les M maxima sont localisés et regroupés en deux modes afin de calculer le seuil.

Les valeurs obtenues des seuils suivant les trois cas forment une matrice de taille $n \times n$ et les valeurs nulles de cette matrice sont recalculées par interpolation puisqu'elles caractérisent des zones homogènes d'un objet ou du fond. La matrice résultante est caractérisée par sa moyenne μ et son écart type σ . Pour valider ou rejeter les différents seuils de la matrice, le critère suivant est utilisé :

$$\begin{aligned} & \text{- Si } t^* \in [\mu - \sigma, \mu + \sigma], \quad t \text{ est conservé} \\ & \text{sinon: } \quad t = 0 \end{aligned} \tag{33}$$

Chaque valeur nulle du seuil est ensuite remplacée par la valeur moyenne des huit voisins non nuls tout en notant que chaque fenêtre est binarisée par la valeur du seuil qui lui correspond.

2.6.2.2 *Seuillage local en maximisant un critère d'homogénéité*

Le seuil local est déterminé automatiquement à l'intérieur de chaque fenêtre en se basant sur la forme des histogrammes et en maximisant un critère d'homogénéité. Ce critère d'homogénéité n'est appliqué que sur des fenêtres qui comportent le fond et l'objet [37]. En mettant en évidence la nature unimodale, bimodale ou multimodale de l'histogramme de chaque fenêtre en utilisant la transformation d'histogrammes des niveaux de gris introduite précédemment, on

peut séparer les fenêtres qui ne comportent que le fond ou l'objet de celles qui comportent le fond et l'objet. A ces dernières, on applique le critère d'homogénéité correspondant à la fonction suivante $F(t)$ pour déterminer le seuil optimal t^* .

$$F(t) = 1 - \frac{\sigma_1^2 + \sigma_2^2}{\sigma_T^2} \quad (34)$$

où l'on voit apparaître les variances des niveaux de gris de chacune des deux classes et la variance totale des niveaux de gris de la fenêtre.

Par ailleurs:

$$\sigma_i^2 = \sum_{(x,y) \in \omega_i} (I_g(x,y) - \mu_i)^2 \quad (35)$$

$$\text{et } \mu_i = \frac{\sum_{(x,y) \in \omega_i} I_g(x,y)}{N_i} \quad (36)$$

où N_i est le nombre de pixels correspondant à la fenêtre W_i , $i=1,2$.

Le seuil optimal est la valeur du niveau de gris qui maximise la fonction d'homogénéité $F(t)$. Chaque fenêtre est binarisée par le seuil qui lui est affecté. Aux fenêtres ne possédant pas de seuil, celui-ci est choisi égal à la moyenne des seuils non nuls des huit fenêtres voisines.

2.6.3. Comparaison entre les méthodes de binarisation

Etant donné qu'il existe une variété de méthodes de binarisation et qu'aucune d'elle n'est universelle comme le précise Rosenfeld [38], au lieu de procéder à ce choix qui, habituellement, se fait sur la base de critères subjectifs telle que l'appréciation visuelle, nous avons utilisé deux critères, l'un basé sur l'uniformité et l'autre sur le contour.

2.6.3.1 Critère d'uniformité

Ce critère d'uniformité est basé sur le calcul de la fonction d'homogénéité telle que:

$$F(t) = 1 - \frac{\sigma_1^2 + \sigma_2^2}{\sigma_T^2} \quad (37)$$

$$\sigma_i^2 = \sum_{(x,y) \in w_i} [L(x,y) - \mu_i]^2 \quad (38)$$

2.6.3.2 Critère du contour

Ce critère de contour est basé sur le calcul du gradient généralement utilisé dans la détection de contour. Il consiste à mesurer la quantité S suivante:

$$S = \frac{\sum_{x,y} \text{Sgn}(L(x,y) - L_M(x,y)) \Delta(x,y) \text{Sgn}(L(x,y) - t)}{\sum_{x,y} \Delta(x,y)} \quad (39)$$

où $\text{Sgn}(\beta) = 1$ si $\beta \geq 0$
 -1 si $\beta < 0$

Dans cette formule $L_M(x,y)$ est la moyenne des niveaux de gris au voisinage $M(x,y)$ du pixel (x,y) et $\Delta(x,y)$, le gradient au point (x,y) .

2.6.3.3 Résultats obtenus

Pour effectuer cette comparaison, plusieurs images de différents objets ont été prises à l'aide d'une caméra CCD interfacée avec une carte d'acquisition d'images, la VIDEOMAKER. Le tableau 3 suivant montre les résultats obtenus avec une image sur laquelle nous avons volontairement introduit un effet d'ombre caractérisée par un histogramme franchement bimodal. D'après ces résultats, les images binaires obtenues sont appréciables pour l'ensemble des méthodes, sauf celles de Pun et celle de la conservation

des moments comme on peut le constater visuellement et à partir des coefficients d'uniformité et de contour. Parmi les méthodes globales, la méthode de maximisation de la variance interclasse a fourni des résultats qui sont meilleurs par rapport aux autres. Les contours obtenus sont bons pour une grande variété d'images.

La méthode utilisant le seuillage par maximisation de l'entropie introduit du biais et, si les images sont bruitées, les contours sont disjoints et la segmentation mauvaise.

Pour la méthode par minimisation de l'erreur quadratique, les contours obtenus ne sont pas très précis.

La méthode par conservation des moments fournit généralement de bons résultats lorsque les images sont contrastées.

Comparativement aux méthodes globales, les méthodes locales sont beaucoup plus performantes (tableau 3). Notons toutefois que le temps de calcul est nettement plus élevé en utilisant les méthodes locales par rapport aux méthodes globales. La méthode la plus lente est de loin la méthode de minimisation de l'erreur quadratique. Ce temps de calcul dépend du nombre de niveaux de gris et de la taille de l'image.

METHODE	SEUIL	TEMPS	UNIFORMITE	CONTOUR	RESULTATS
Pun	157	0.006	0.663	0.620	MAUVAIS
Max d'Entropie	143	3.07	0.871	0.996	MOYENS
Conser, des Moments	158	0.005	0.601	0.530	MAUVAIS
Var Interclasses	112	0.44	0.920	0.986	BONS
Err Quadratique	146	3.35	0.843	0.986	MOYENS
Test Validation	-	0.10	0.904	0.989	BONS
Critère d'homogénéité.	-	2.08	0.902	0.986	BONS

Tableau 3 : Comparaison des résultats obtenus par différentes méthodes de binarisation pour une image

La figure 4 montre quelques exemples d'images binarisées.

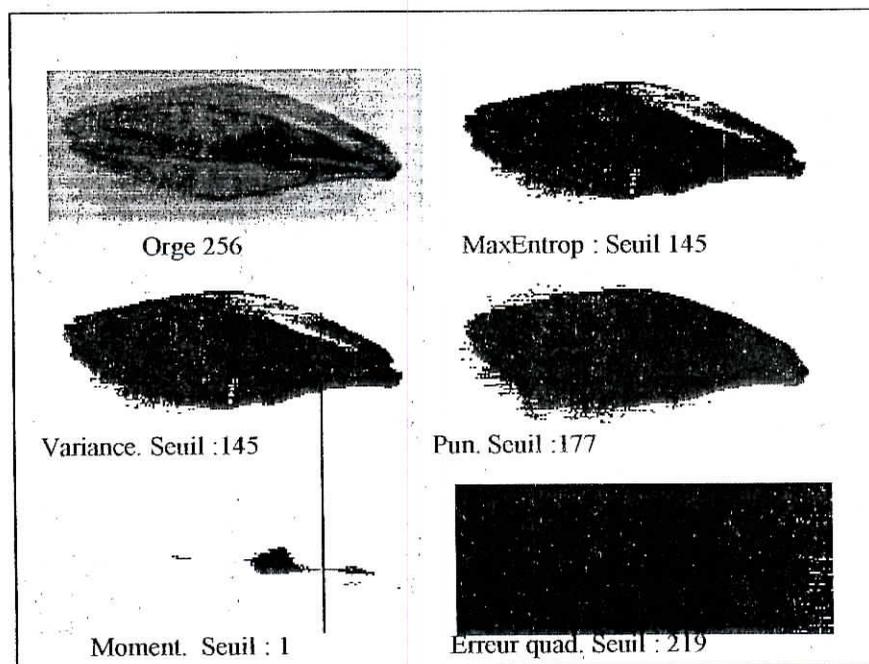


Fig.4 : Exemples d'images binarisées

2.7. Suivi de contours

Les contours peuvent être détectés par un examen systématique des variations locales du niveau de gris au voisinage de chaque point de l'image. Les méthodes utilisées permettent de distinguer les points susceptibles d'appartenir à ces contours [39]. Elles sont classées en nombre de deux :

- les différentielles,
- les suivis de contours .

Dans le premier cas, un opérateur différentiel est utilisé et un point de transition local est obtenu par le calcul de la dérivée locale par rapport à ses voisins. Plusieurs méthodes basées sur le calcul du Gradient et du Laplacien ont été proposées [40]. Avec ces méthodes, les contours obtenus sont souvent épais et doivent être suivis de l'opération de squelettisation (fig.5). Nous tenons aussi à noter que ces opérations coûtent beaucoup en temps de calcul.

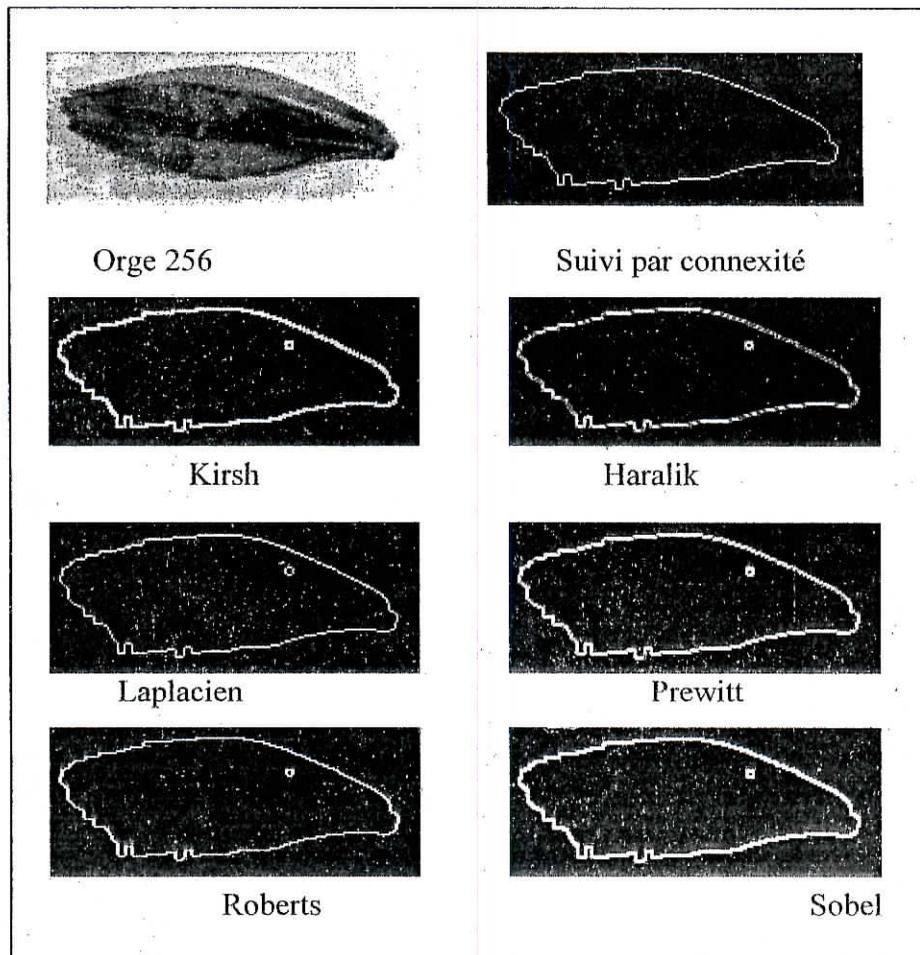


Fig.5 Exemples d'extraction de contours

Dans le deuxième cas, le principe consiste à détecter un point qui appartient au contour et qui sera un point de départ pour la procédure de détection des autres points. Pour ce faire, on teste les huit voisins du point courant du contour jusqu'à trouver le prochain point du contour en explorant l'image ligne par ligne. On s'arrête au premier point du contour. Cette méthode présente l'avantage de pouvoir mémoriser les points du contour. Cette mémorisation est réalisée soit en coordonnées, soit en une suite d'orientations par la chaîne de codage de Freeman (fig.6).

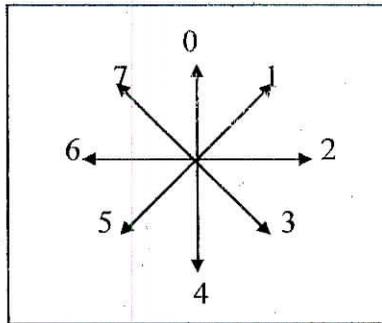


Fig. 6 : Les huit directions de Freeman

La détection du contour par la méthode différentielle demande un temps d'exécution considérable et le contour affiché n'est pas optimal. Il est souvent brouillé, ce qui va compliquer les algorithmes d'extraction des paramètres et risqueraient de fausser les résultats. Par contre, la méthode de suivi de contours appliquée à l'image binarisée donne de meilleurs résultats. Dans notre application, le contour représente notre outil de base à partir duquel a été faite l'extraction de tous les paramètres, à l'exception du centre d'inertie et surface. Par la méthode de suivi de contours par connexité, le contour est plus net, moins épais et le temps d'exécution est relativement réduit. Cette opération nécessite souvent d'autres opérations de lissage binaire pour supprimer des points isolés ou rattacher des points non connexes du contour.

2.8 Conclusion

Après avoir montré que certaines opérations de traitement d'images sont parfois indispensables pour réussir l'opération de reconnaissance des formes, nous allons procéder à l'opération d'extraction des paramètres morphologiques de graines de semence. Nous rappelons toutefois que parmi ces opérations, les plus utilisées sont la binarisation et le suivi de contours. Dans le cas où la qualité des images n'est pas très bonne, on fait ces opérations de filtrage et de lissage.

Chapitre 3

Reconnaissance de Graines à partir des Paramètres Morphologiques

3.1 Introduction

Dans ce chapitre, il s'agit de l'application du processus de reconnaissance de graines. Ce cas pratique va nous permettre de suivre le déroulement des différentes opérations qui y interviennent depuis l'acquisition des images jusqu'à la reconnaissance proprement dite.

3.2 Acquisition des images des quatre espèces de graines

La première opération consiste à saisir en niveaux de gris des images séparées de quatre cents graines de quatre espèces à l'aide d'une caméra CCD en liaison avec un Pentium via une carte de type Vidéo-Maker . Il s'agit de graines d'avoine, de blé, d'orge et de lentille (fig.1). Le choix de ces graines est motivé par le fait que les classes des lentilles et du blé peuvent être facilement séparées. Par contre, visuellement, la ressemblance entre l'orge et l'avoine peut conduire, *a priori*, à un chevauchement des deux classes. L'utilisation de ces quatre espèces de graines va nous permettre d'évaluer les performances de la méthode à utiliser pour les séparer automatiquement puisqu'il s'agit d'espèces proches par leurs formes, couleurs et textures. L'autre motif important est directement lié à l'intérêt de l'application proprement dite. En effet, comme nous l'avons déjà expliqué en introduction générale, pour déterminer l'indice de pureté d'un chargement de graines de semence, on prélève un échantillon de ces graines pour le faire étudier et manipuler par un personnel qui s'appuie sur des documents photographiques avec des risques sur ses fonctions visuelles et des troubles respiratoires à cause des poussières et des produits chimiques contenus dans ces graines. Ceci nous a motivés à procéder à l'automatisation complète de cette opération en introduisant des techniques de traitement d'images et de reconnaissance des formes.

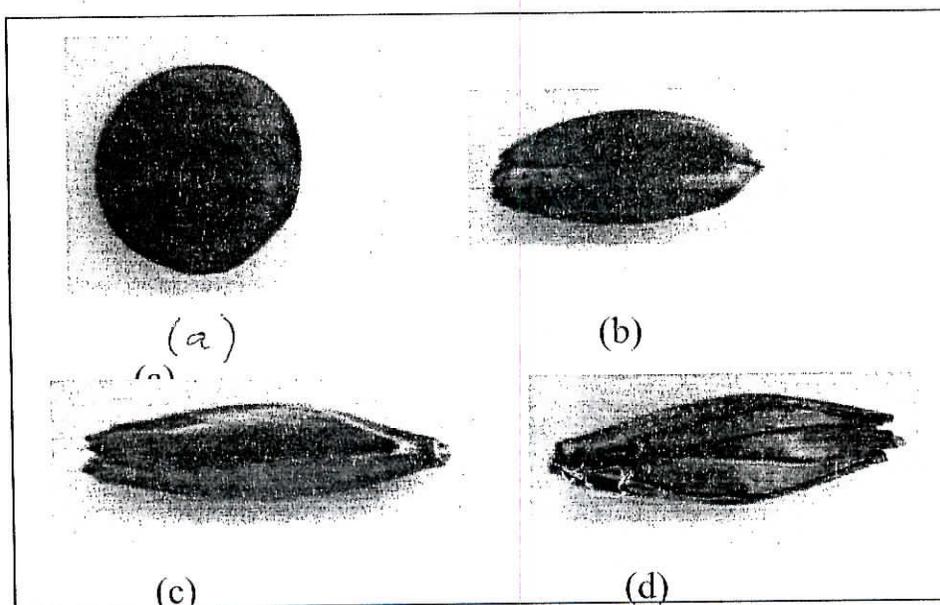


Fig.1 Images des quatre espèces de graines

(a) : lentille,

(b) : blé,

(c) : avoine,

(d) : orge.

3.3. Binarisation des images des différentes graines

Avant la phase d'extraction des paramètres morphologiques, l'image de l'objet est séparée du fond sur lequel il est posé en utilisant une des méthodes de binarisation. Rappelons que l'efficacité de certaines de ces méthodes par rapport aux autres dépend de la qualité de l'image elle-même (fig.2)

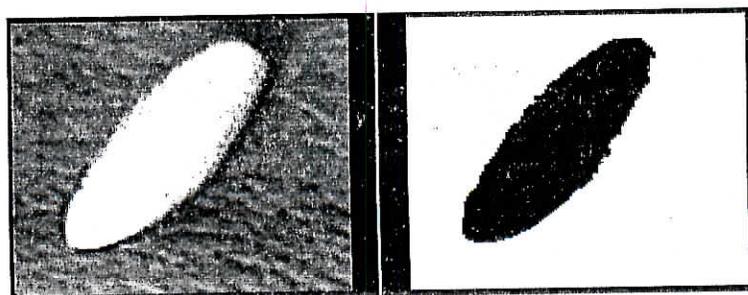


Fig.2. : Exemple de binarisation de l'image d'une graine

Notons par ailleurs, que des images comprenant plusieurs graines en même temps peuvent aussi être saisies. Dans ce cas, il se posera le problème de séparation des images des graines. S'il arrive que les images soient entachées d'un bruit, un filtrage de l'image s'avère nécessaire. Ce filtrage doit toutefois être utilisé avec prudence au risque de faire disparaître des informations importantes très caractéristiques de l'objet à reconnaître.

3.4 Suivi de contour

Aux images binarisées obtenues, on applique un suivi de contour par connexité pour obtenir les contours fermés des graines comme le montre l'exemple de la figure 3. Par rapport aux contours obtenus par les opérateurs classiques dont quelques exemples sont montrés au chapitre 2, le résultat obtenu par suivi de contours par connexité est plus net, moins épais et plus uniforme.

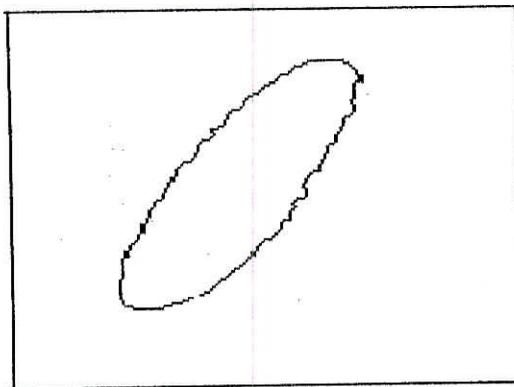


Fig.3 : Exemple d'un suivi de contour par connexité

3.5 Calcul des paramètres morphologiques

A partir des images binarisées et leurs contours, treize paramètres morphologiques invariants par translation ou rotation ont été calculés. Il s'agit des paramètres suivants :

- *le périmètre* : c'est la somme des segments de droites reliant les points consécutifs du contour. Le calcul du périmètre peut être effectué à partir du tableau de coordonnées résultant du suivi de contour par connexité. Il peut aussi être évalué par le nombre de pixels le constituant. Son unité de mesure est alors en nombre de pixels.

- *la surface* : c'est le nombre de pixels constituant l'image binaire. L'unité de mesure de la surface est évaluée en nombre de pixels.

- *la compacité* : elle est comprise entre 0 et 1 et calculée par la formule :

$$C = \frac{4\pi S}{P^2} \quad (1)$$

où S est la Surface et P le périmètre. On notera que pour un cercle, la compacité est égale à l'unité.

- *le centre de gravité* : c'est le point G(x_g,y_g) de coordonnées X_g et Y_g qui sont, respectivement, la moyenne des abscisses et la moyenne des ordonnées des points du contour de l'image.

- *le centre d'inertie* : c'est le point I de coordonnées X_i et Y_i qui sont respectivement la moyenne des abscisses et la moyenne des ordonnées des points de la surface de l'image binarisée.

- *les distances* : les quatre distances suivantes ont été calculées. Il s'agit de :

- la plus grande distance du centre de gravité au contour (GA)
- la plus petite distance du centre de gravité au contour (GB)
- la plus grande distance du centre d'inertie au contour (IC)
- la plus petite distance du centre d'inertie au contour (ID)

Pour chaque distance, on calcule son opposée à savoir la distance du centre de gravité G ou d'inertie I au point du contour se situant sur la même droite passant par le centre de gravité ou d'inertie (fig.4). Nous obtenons respectivement :

- la distance (GA') opposée à la plus grande distance (GA) de G au contour.
- la distance (GB') opposée à la plus petite distance (GB) de G au contour.
- la distance (IC') opposée à la plus grande distance (IC) de I au contour
- la distance (ID') opposée à la plus petite distance (ID) de I au contour

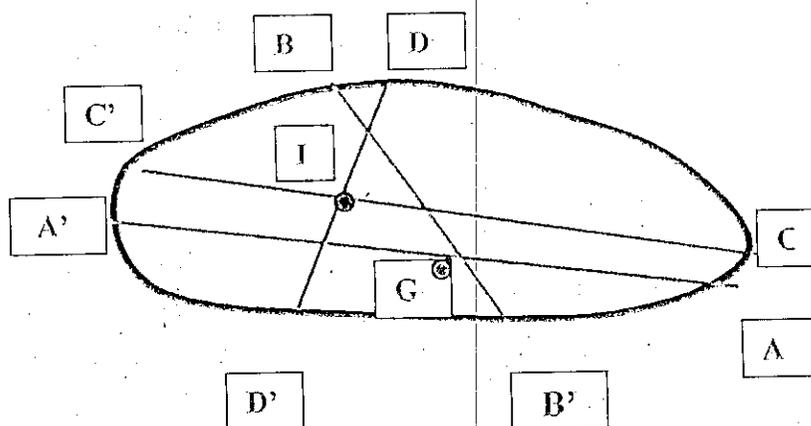


Fig.4 : Les différentes distances calculées

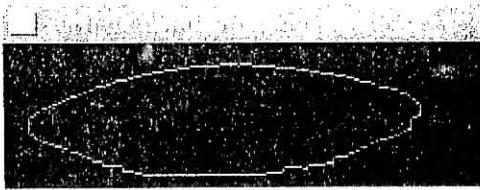
Les moments invariants suivants peuvent aussi être calculés :

$$\psi_1 = \frac{\mu_{20}}{\mu_{00}^2} + \frac{\mu_{02}}{\mu_{00}^2} \quad \text{et} \quad \psi_2 = \frac{\mu_{20}^2 + \mu_{02}^2}{(\mu_{00}^2)^2} + 4 \frac{\mu_{11}^2}{\mu_{00}^2} \quad (2)$$

où :

$$\mu_{pq} = \frac{I}{N} \sum_{i=1}^N (x_i - \bar{x})^p (y_i - \bar{y})^q \quad (3)$$

Les résultats de ces calculs sont présentés sous forme d'un tableau à 400 lignes et onze colonnes. Les centres de gravité et d'inertie qui sont des paramètres de position ne sont pas introduits dans ce tableau. Le tableau 1 montre un exemple de valeurs en pixels des paramètres calculés.



Paramètre	Valeur (en pixels)
Périmètre P	456
Surface S	9087
Compacité C	0.55
Distance GA	98
Distance GA'	97
Distance GB	27
Distance GB'	27
Distance IC	99
Distance IC'	95
Distance ID	25
Distance ID'	30

Tableau 1 : Exemple de valeurs des paramètres morphologiques calculés

3.6 Normalisation et Réduction des paramètres morphologiques

3.6.1 Normalisation

Sachant que les grandeurs des paramètres considérés peuvent être de natures différentes et de valeurs très faibles pour certaines ou trop élevées pour d'autres, leur normalisation est nécessaire avant d'effectuer l'opération de classification. Pour ce faire, la formule suivante a été appliquée :

$$x_{ij} = \frac{x_{ij} - x_{\min}}{x_{\max} - x_{\min}}, \quad i = 1, 2, \dots, n \quad \text{et} \quad j = 1, 2, \dots, m \quad (4)$$

où :

- x_{ij} est la valeur du paramètre j de l'objet i,
- x_{\min} la valeur minimale du paramètre j
- x_{\max} la valeur maximale du paramètre.

3.6.2. Réduction des paramètres

Parmi ces paramètres, seuls ceux qui sont porteurs d'informations sont à prendre en considération. Ceci peut être effectué par application de l'Analyse en Composantes Principales (ACP). On rappelle que l'ACP recherche les axes principaux d'inertie. Les individus peuvent ainsi être présentés sur un des plans de l'espace multidimensionnel formés par deux axes principaux. La figure 5 montre un exemple de projection des quatre cents objets selon deux axes principaux.

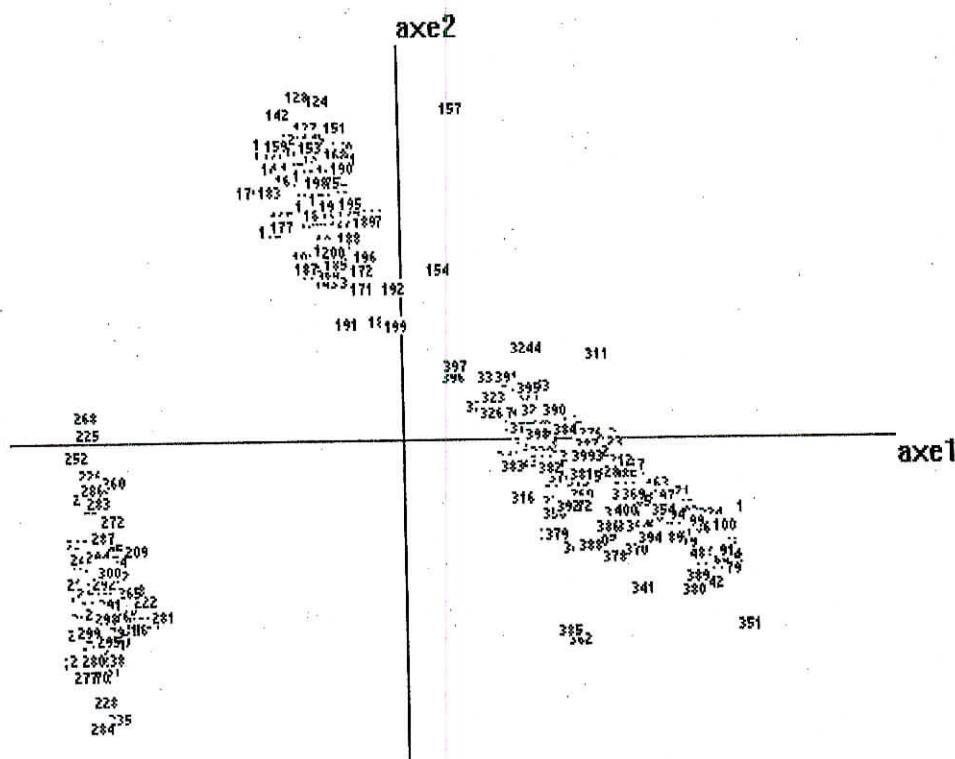


Fig.5 : Projection des 400 objets selon deux axes principaux

Les pourcentages d'inertie expliqués par six axes sont :

Axe	1	2	3	4	5	6
% d'inertie	54.52	33.88	6.67	1.72	1.27	0.75
Total	98.81					

Ainsi, les dimensions initiales 400x11 du tableau de données sont réduites finalement à 400x6.

3.7. Classification automatique des données

Pour classer nos données, même si nous connaissons les classes *a priori*, la méthode d'agrégation autour des centres mobiles citée précédemment est utilisée. La convergence de l'opération est rapide puisque nous connaissons les partitions initiales. Seulement, pour rendre la méthode plus autonome vis à vis de l'utilisateur, nous avons préféré l'associer à l'A.C.P. puisque celle-ci nous fournit déjà une visualisation graphique qui va nous permettre de fixer la partition initiale en pointant et cliquant seulement la souris sur le centre de ces classes.

3.7.1. Association avec l'A.C.P.

La visualisation des regroupements des objets selon les axes principaux d'inertie de l'A.C.P. permet donc d'enclencher la classification par agrégation autour des centres mobiles après avoir fixé les centres des partitions initiales. On remarque que le temps d'exécution est réduit par rapport au cas habituel où l'Agrégation Autour des Centres Mobiles est utilisée seule.

3.7.2 Utilisation de la méthode Mixte

On rappelle que dans la méthode mixte décrite précédemment et dont le principe consiste en un déplacement d'un observateur fictif dans l'espace multidimensionnel des données, la visualisation dans leur contexte de regroupement en classes de ces données peut s'avérer suffisante pour la classe des lentilles et celle du blé. Par contre, pour la classe d'orge et d'avoine qui se chevauchent, on doit fixer les partitions initiales en cliquant sur les centres de ces regroupements puis faire appel à la méthode d'Agrégation autour des Centres Mobiles. La figure 6 montre une visualisation graphique des quatre classes obtenues.

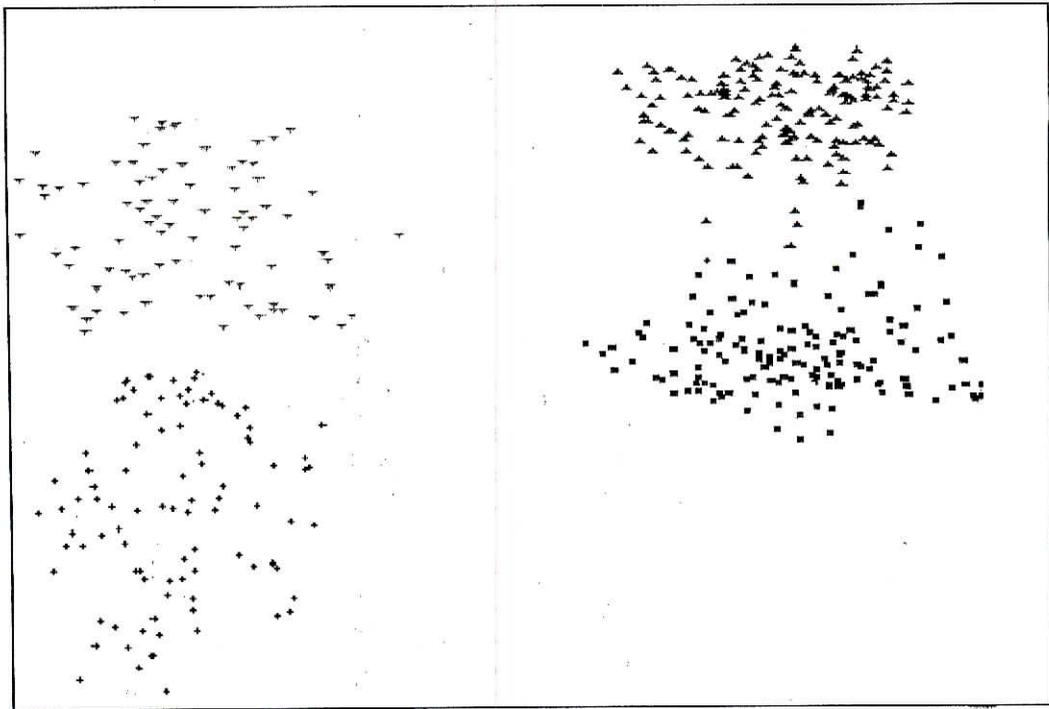


Fig.6 :Visualisation graphique des quatre classes des graines

3.7.3 Résultats obtenus après la classification

L'application de la méthode d'agrégation autour des centres mobiles a abouti à la formation des quatre classes suivantes :

La première classe est formée des cent onze éléments suivants :

1 4 5 6 7 9 10 12 13 14 15 16 17 18 19 20 21 22 25 26 27 28 29 30 31 32
 33 34 35 36 37 39 40 42 43 44 45 46 47 48 49 50 51 52 53 55 57 59 60
 61 62 63 64 66 67 70 71 72 74 75 76 77 78 79 80 81 82 83 84 85 86 87
 88 89 91 92 93 94 95 97 100 302 307 308 309 314 315 330 336 339 341
 344 347 351 352 354 356 362 363 364 365 369 370 378 380 385 386
 388 389 394 400.

La deuxième classe est formée des cents éléments suivants :

101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116
 117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132
 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148
 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164
 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180
 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196
 197 198 199 200.

La troisième classe est formée des cents éléments suivants :

201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216
217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232
233 234 235 236 237 238 239 240 241 242 243 244 245 246 247 248
249 250 251 252 253 254 255 256 257 258 259 260 261 262 263 264
265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280
281 282 283 284 285 286 287 288 289 291 292 293 294 295 296 297
298 299 300.

La quatrième classe est formée des quatre vingt neuf éléments suivants :

2 3 8 11 23 24 38 41 54 56 58 65 68 69 73 90 96 98 99 301 303 304 305
306 310 311 312 313 316 317 318 319 320 321 322 323 324 325 326
327 328 329 331 332 333 334 335 337 338 340 342 343 345 346 348
349 350 353 355 357 358 359 360 361 366 367 368 371 372 373 374
375 376 377 379 381 382 383 384 387 390 391 392 393 395 396 397
398 399.

Ces résultats montrent que les graines de blé et de lentille ont été bien séparées (classe 2 et 3). Par contre, les classes "orge" et "avoine" se chevauchent avec un taux de confusion voisinant les 7,5%. En effet, dans la classe "avoine", 81 éléments sont reconnus et, à cette classe il manque les éléments 2, 3, 8, 11, 23, 24, 38, 41, 54, 56, 58, 65, 68, 69, 73, 90, 96, 98, 99 que nous retrouvons dans la classe "orge". Dans cette même classe "avoine", on trouve 30 éléments supplémentaires qui appartiennent en réalité à la classe "orge". Il s'agit des éléments 302, 307, 308, 309, 314, 315, 330, 336, 339, 341, 344, 347, 351, 352, 354, 356, 362, 363, 364, 365, 369, 370, 378, 380, 385, 386, 388, 389, 394 et 400.

Inversement, dans la classe "orge", 70 éléments de cette classe sont bien classés. Dans cette classe on retrouve les 19 éléments 2, 3, 8, 11, 23, 24, 38, 41, 54, 56, 58, 65, 68, 69, 73, 90, 96, 98, 99. A cette classe "orge", il manque les 30 éléments supplémentaires placés dans la classe "avoine".

3.8 SIGNATURE POLAIRE

3.8.1 Calcul de la signature polaire

Pour chacune des quatre classes formées, la signature polaire moyenne a été calculée. On rappelle que la signature polaire d'une graine est la fonction $\rho(\theta)$ représentant la distance du centre de gravité $G(X_g, Y_g)$ à un point du contour en fonction de θ . Le point de départ pour cet angle peut être fixé comme étant le point A le plus éloigné de $G(X_g, Y_g)$ (fig.7). On a ainsi :

$$\theta_i = \text{Arctg}\left(\frac{y_i - y_l}{x_i - x_l}\right) \quad (5)$$

$$\rho(\theta_i) = \sqrt{(x_i - x_l)^2 + (y_i - y_l)^2} \quad (6)$$

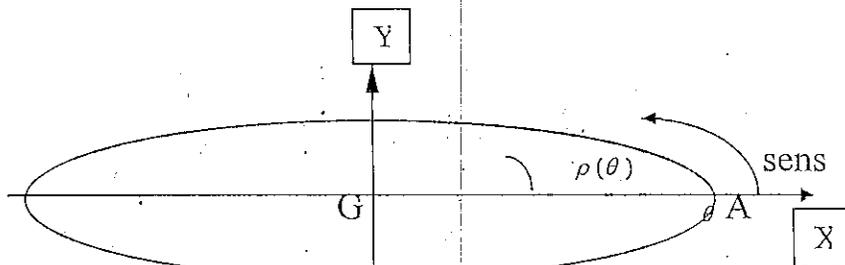


Fig.7 : Calcul de la signature polaire

En ce qui concerne l'invariance en rotation de la signature polaire, l'angle $\theta=0$ correspond au point A le plus éloigné du centre de gravité. Elle peut aussi être réglée par une rotation de tous les objets concernés par la signature polaire moyenne jusqu'à ce que la distance euclidienne entre deux vecteurs représentant les signatures polaires de deux objets soit minimale.

3.8.2 Modèle d'apprentissage

Sachant que la signature polaire est une caractéristique synthétique représentant et englobant la majorité des paramètres déterminés, le choix de cette signature polaire paraît très intéressant comme modèle de

référence du fait qu'elle est invariante par translation. Ainsi, la dernière phase du processus décisionnel consistera à identifier un élément inconnu à partir de son image en se basant sur la seule notion de signature polaire moyenne de chaque classe formée. Cette signature polaire moyenne correspond à la moyenne des signatures de chaque classe et est calculée par l'expression suivante : Pour chaque classe k , la signature polaire moyenne est calculée par l'expression suivante :

$$\rho_m^k(\theta_i) = \frac{1}{N_k} \sum_{j=1}^{N_k} \rho_j^k(\theta_i) \quad (7)$$

où N_k est le nombre de graines de la classe k et $\rho_j^k(\theta_i)$ la valeur du rayon vecteur de la graine j correspondant au $i^{\text{ème}}$ angle polaire θ_i (fig.8).

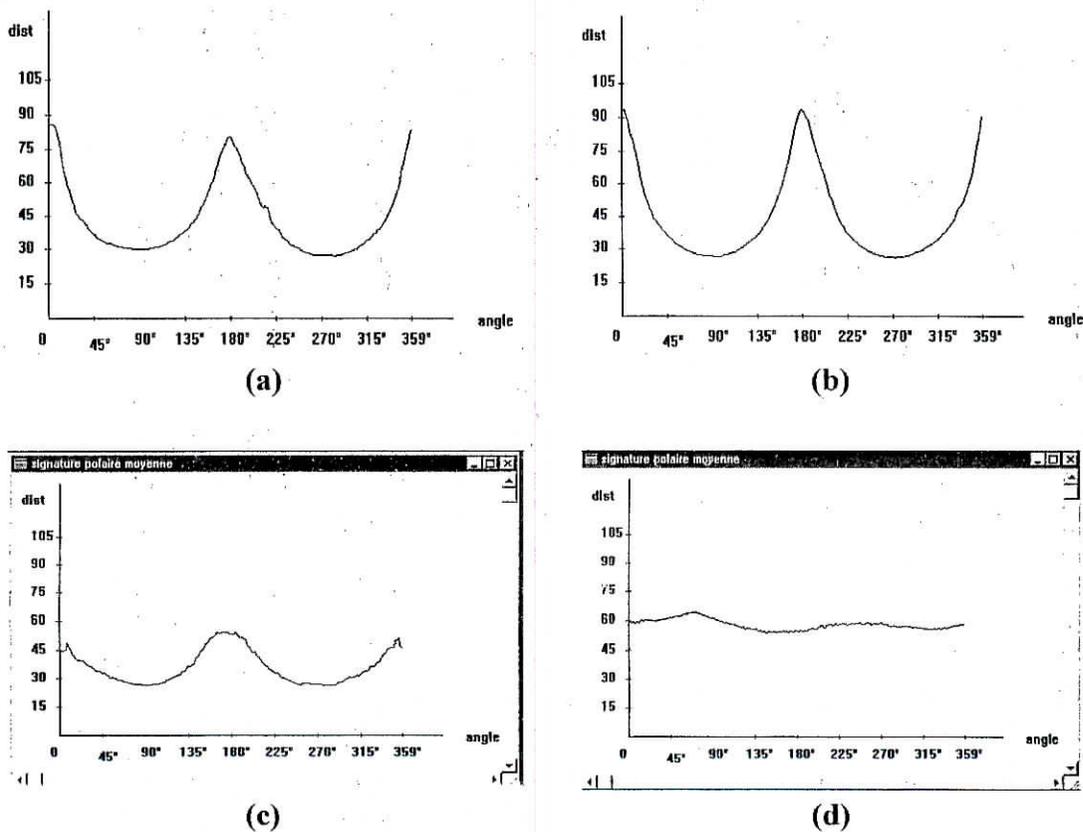


Fig.8 : Signatures polaires moyennes :

- (a) : de la classe d'orge
- (b) : de la classe de l'avoine
- (c) : de la classe du blé
- (d) : de la classe des lentilles

Dans la pratique, cette signature polaire peut aussi être représentée par un vecteur à 360 éléments correspondant aux valeurs de $\rho(\theta)$ calculée pour chaque degré et pour tout le contour fermé dans le cas où il n'existe pas de contours concaves. En effet, dans le cas où l'on a une forme concave ou creuse, à l'angle θ correspondront plusieurs rayons ρ . Ce problème peut être réglé par une technique de représentation se basant sur l'abscisse curviligne de la tangente par rapport aux différents points du contour de l'objet en question (fig.9).

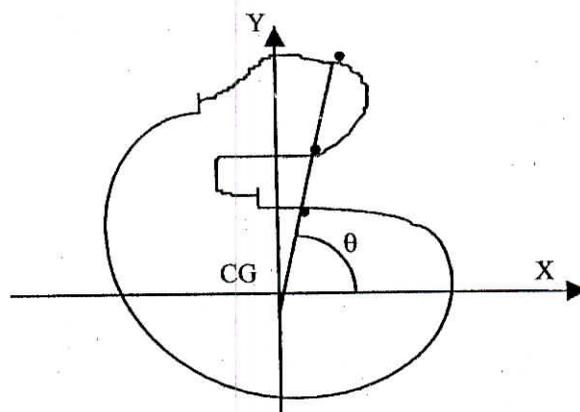


Fig. 9 : Signature polaire d'une forme concave

3.9 Phase de reconnaissance

La reconnaissance proprement dite consiste donc en l'affectation d'un nouvel objet à la classe avec laquelle il présente des caractéristiques semblables. Ceci se fait en calculant l'erreur quadratique entre la signature polaire de l'image de la graine à reconnaître et les quatre

signatures polaires moyennes, soit :

$$e(k) = \sqrt{\sum_{i=1}^{360} (\rho_m^k(i) - \rho_x(i))^2} \quad (8)$$

où $\rho_m^k(i)$ est la $i^{\text{ième}}$ composante de la signature polaire moyenne de la $k^{\text{ième}}$ classe et $\rho_x(i)$ la $i^{\text{ième}}$ composante du tableau de la signature polaire de l'objet à reconnaître. La graine à identifier est affectée à la classe qui correspond à l'erreur quadratique la plus faible.

3.10 Logiciel

Réalisé en C++ , le logiciel existe sous deux formes.

Sous la première, toutes les opérations apparaissent dans le menu afin de suivre le déroulement du programme (fig.10).

Sous sa deuxième forme, le logiciel est destiné à automatiser complètement la procédure (fig.11). En effet, si dans ce travail les images sont chargées à partir d'un fichier du disque dur, dans le projet en question, il s'agira de ramener automatiquement l'objet sous la caméra. Le processus de reconnaissance se déclenchera ensuite. Pour ce faire, il y a lieu de maîtriser la partie hardware en ce qui concerne les lignes de contrôle de la carte d'acquisition d'images utilisée.

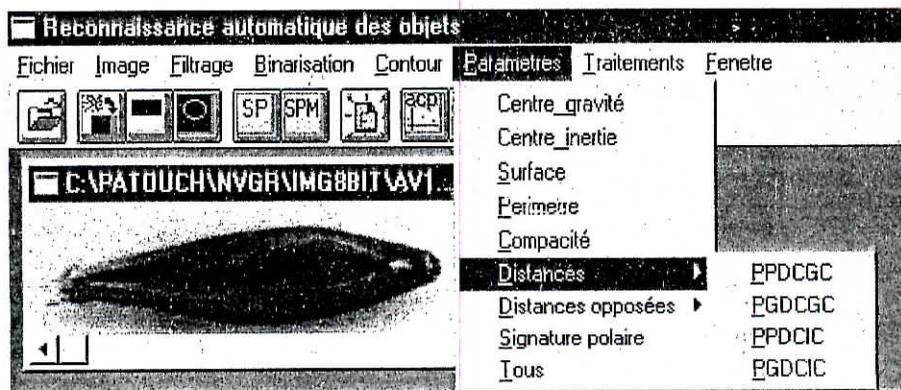
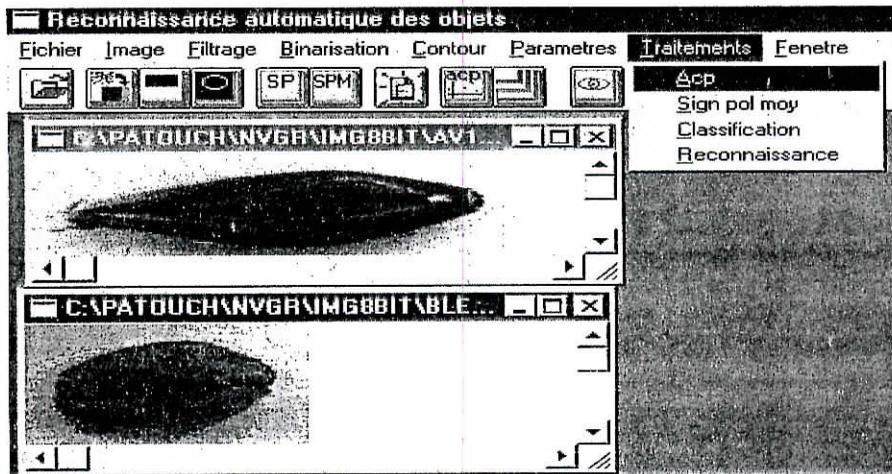


Fig.10 : Partie interactive du logiciel interactif réalisé

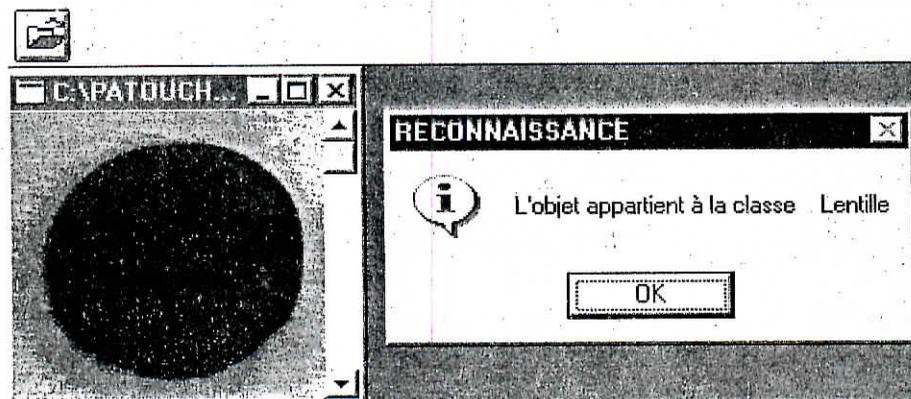


Fig.11 : Partie automatique du logiciel

3.11 Conclusion

Si les étapes de la procédure décrite dans ce chapitre peuvent être appliquées à divers autres types d'objets, on doit rappeler que dans certaines situations, l'éclairage mal adapté peut entraîner une mauvaise binarisation. Ceci influera sur la reconnaissance proprement dite. Par ailleurs, on devra signaler que l'application des différents opérateurs d'extraction de contours ne fournit pas souvent de bons résultats vue l'épaisseur des contours obtenus et le temps de calcul lié à leur exécution. Les paramètres de couleur telles que la teinte, la saturation et la luminance peuvent aussi être utilisés dans ce type particulier d'application. Il serait aussi intéressant d'extraire les paramètres de texture au détriment d'un temps de calcul élevé. La méthode de décision basée sur la détermination de l'erreur quadratique moyenne entre signatures polaires donne de très bons résultats. Cependant, la notion de classification d'une observation inconnue sur la base de la distance minimale euclidienne par rapport aux centres de gravités de toutes les classes peut aussi être utilisée. Notons que le calcul des moments interclasses peut être utilisé pour juger de la qualité de la classification. Plus la valeur du moment interclasse est grande, plus les classes sont éloignées les unes des autres. Plus la valeur du moment intraclasse est réduite, meilleure est la concentration des individus autour de leurs centres de gravités respectifs. Une bonne classification signifie donc un rapport maximal entre le moment interclasse et le moment total.

Chapitre 4

**Reconnaissance de Formes à l'aide
des paramètres de texture et de Réseaux de Neurones**

4.1 Introduction

Dans ce chapitre, nous présentons une méthode de reconnaissance automatique des graines basée sur l'analyse des paramètres de texture calculés à partir de la matrice de cooccurrence et d'un réseau de neurones multicouches. L'apprentissage est réalisé à l'aide d'un algorithme de rétropropagation.

4.2 Rappel sur les réseaux de neurones

Défini comme étant l'unité fonctionnelle de base du système nerveux, le neurone biologique constitue un système dynamique bouclé, non linéaire, asynchrone au pouvoir très important qu'est le cerveau. Ainsi, dans le but d'obtenir des performances et des caractéristiques proches de celles du cerveau humain, les chercheurs se sont mis à la modélisation mathématique du neurone biologique. Ainsi, le premier neurone artificiel de l'histoire a été présenté en 1943 par Mc Culloch sous forme d'un produit scalaire entre un vecteur d'entrée et un vecteur poids suivi d'un élément à seuil [41]. En 1950, Rosenblat mis au point le "Perceptron" qui diffère du modèle de Mc.Culloch et Pitts par sa faculté d'apprentissage lui permettant ainsi de prendre de bonnes décisions dans les problèmes de classification [42]. Des travaux basés sur la règle de Hebb, publiés dans les années 1970, ont apporté des modèles intéressants mais ils étaient bloqués par les problèmes d'instabilité et de concepts liés aux systèmes dynamiques [43]. En 1986, Rumelhart, Hinton et Williams ont proposé l'algorithme de rétropropagation pour traiter les problèmes non linéaires [44]. Ainsi, à cet algorithme, sont venus s'ajouter plusieurs autres, inspirés, parfois de celui-ci [45,46].

4.2.1 Neurone formel et Fonction d'activation

Tout d'abord, rappelons que le neurone biologique est composé :

- de la *dendrite* qui collecte les signaux venant d'autres cellules ou de l'extérieur, au niveau des points de contacts appelés « *Synapses* »,
- du *soma* qui est le corps cellulaire vers où les informations recueillies par les synapses sont acheminées,
- de l'*axone* qui est la partie qui s'occupe de la transmission de l'information issue du corps cellulaire vers les neurones avoisinants.

Inspiré du neurone biologique, le neurone formel est un modèle mathématique ou processeur élémentaire qui, en recevant un signal X à une ou plusieurs composantes sur ses connexions entrantes, calcule la somme pondérée de ces entrées et détermine sa sortie Y en fonction de son activité et éventuellement d'un seuil (fig.1).

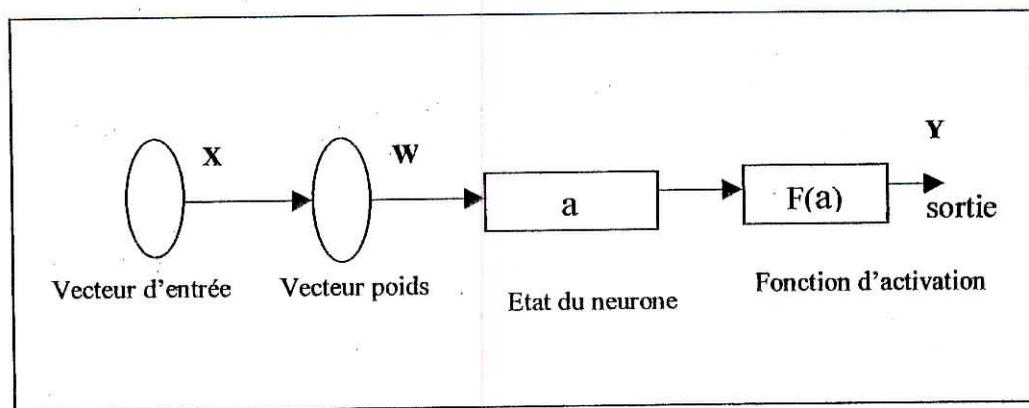


Fig.1. Modèle général d'un neurone formel

Ainsi, on a :

- le vecteur d'entrée (x_1, x_2, \dots, x_n) ,
- la matrice des poids $(w_{i1}, w_{i2}, \dots, w_{ij}, \dots, w_{in})$,
- la fonction d'entrée qui peut être :

- linéaire $a = \sum_{j=1}^n w_{ij} x_j$
- seuillée $a = \sum_{j=1}^n w_{ij} x_j - \theta$, θ étant le seuil,
- polynomiale $a = \left(\sum_{j=1}^n w_{ij} x_j \right)^n$

- la fonction d'activation qui définit l'état interne du neurone $F(a)$
- la sortie Y qui est une fonction identité telle que $Y=F(a)$

Les fonctions d'activation $F(a)$ jouent un rôle important dans le comportement du neurone et prennent différentes formes selon l'utilisation du réseau. Parmi ces fonctions, la plus utilisée est la sigmoïde qui est différentiable et permet une sortie comprise dans l'intervalle $[0,1]$ et donnée par la formule :

$$f(x) = \frac{1}{1 + e^{-\beta x}} \quad (1)$$

Par ailleurs, les réseaux de neurones peuvent être répartis en deux grandes catégories selon qu'ils interdisent ou permettent les liens récurrents entre les neurones. On distingue, d'abord, les *Réseaux de neurones statiques* ou *non bouclés* dans lesquels il n'existe pas de « feedback » ou retour d'information et les *Réseaux de neurones dynamiques* ou *bouclés* dans lesquels l'introduction de feedback entre les neurones rend le réseau dynamique [47].

4.2.2 Réseaux statiques ou non bouclés

Un réseau de neurones non bouclé est représenté graphiquement par un ensemble de neurones connectés entre eux, l'information circulant des entrées vers les sorties sans retour en arrière. Ces réseaux voient l'activation de leurs neurones les traverser en une vague unique et n'ont pas véritablement de comportement dans le temps. Ils peuvent être utilisés pour les problèmes de

classification ou d'approximation de fonctions non linéaires complexes. Parmi ces réseaux, on peut citer le *Perceptron* proposé par Rosenblatt en 1959 et l'*AdaLinE* « Adaptive Linear Element » proposé par Widrow et Hoff en 1960 [48,49].

Dans leurs formes simples, les perceptrons sont généralement constitués d'une couche d'entrée à n éléments reliés à une couche cachée à m cellules d'associations ou masques ou unités ϕ_i et une sortie binaire à seuil avec comme fonction d'activation F , la fonction de Heaviside.

$$Y = F(a) = F\left(\sum_{j=1}^n w_{ij} \phi_j + \theta\right) \quad \text{avec } \theta=0 \quad (2)$$

Selon la nature des fonctions ϕ_i , les perceptrons peuvent être regroupés en différentes familles comme cela est montré dans [50]. Quant à l'algorithme d'apprentissage du perceptron, il est semblable à celui de Hebb à la différence que la modification des poids à l'instant $t + \delta t$ suit l'équation (3) pour le premier et l'équation (4) pour le second:

$$W_{ij}(t + \delta t) = W_{ij}(t) + \mu[(d - y)x_i] \quad (3)$$

$$W_{ij}(t + \delta t) = W_{ij}(t) + \mu x_i x_j \quad (4)$$

où $W_{ij}(t)$ est le poids de la connexion entre le neurone j d'activation x_j et le neurone i d'activation x_i à l'instant t , μ un paramètre positif de l'intensité d'apprentissage, y la sortie obtenue et d la sortie désirée. Quant à la différence entre la règle du perceptron et celle de Widrow-Hoff, elle réside au niveau du calcul de la modification des poids. Dans le cas du perceptron, si :

- $d=y$, les poids ne sont pas modifiés,
- $d-y=1$, le poids se transforme en $W_{ij} + \mu x_i$
- $d-y=-1$, le poids se transforme en $W_{ij} - \mu x_i$, x_i étant les entrées seuillées,

Dans le cas de Widrow-Hoff, le signal d'erreur est égal à la différence entre la somme pondérée des entrées non seuillées et le résultat attendu, ie. :

$$W_{ij}(t + \delta t) = W_{ij}(t) + \Delta W_{ij}(t) \quad (5)$$

avec $\Delta W_{ij}(t) = \mu x_i(d_i - y_i)$, règle qui effectue une descente du gradient sur une mesure d'erreur quadratique : $E = 1/2 \sum (d_i - y_i)^2$.

Dans le cas de l'AdaLinE, l'état du neurone est donné par :

$$a = \sum_j W_{ij}(t) X_j + \theta \quad (6)$$

où θ son seuil et X_j l'entrée extérieure j (fig.2).

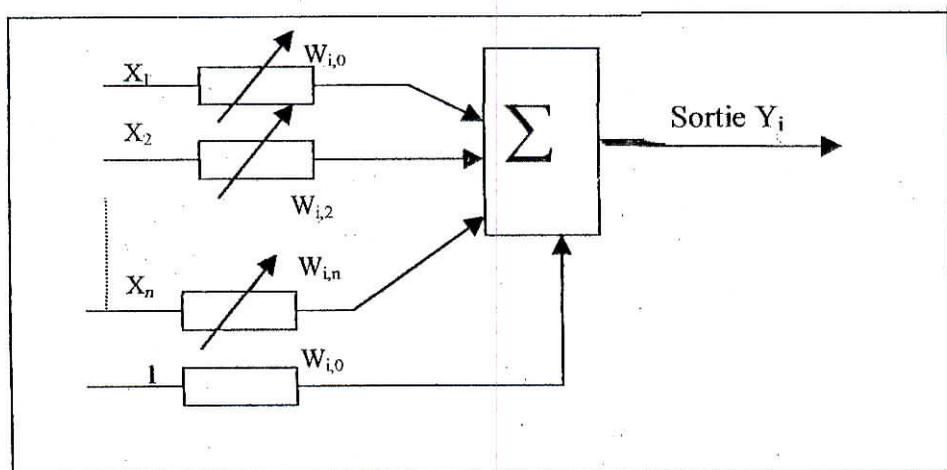


Fig.2 : Schéma du réseau Adaline

4.2.3 Réseaux dynamiques ou bouclés

Contrairement aux réseaux de neurones non bouclés, les réseaux de neurones bouclés peuvent avoir une topologie de connexions quelconque, comprenant, notamment, des boucles qui ramènent, aux entrées, la valeur d'une ou plusieurs sorties. Pour la causalité du système, un retard doit être associé à chaque boucle. Ainsi, un réseau de neurones bouclé qui est souvent à temps discret est régi par une ou plusieurs équations aux différences non linéaires, résultant de la composition des fonctions réalisées par chacun des

neurones et des retards associés à chacune des connexions. La structure dynamique de ces réseaux est gouvernée par l'équation :

$$\alpha_0 \dot{x}_i(t) + \alpha_1 x_i(t) = v_i(t) \quad (7).$$

Cependant la forme la plus générale appelée forme canonique est donnée par :

$$x(k+1) = \varphi[x(k), u(k)] \quad \text{et} \quad y(k) = \psi[x(k), u(k)] \quad (8)$$

où φ et ψ sont des fonctions non linéaires réalisées par un réseau de neurones non bouclé et k le temps discret.

Tout réseau de neurones peut être mis sous la forme canonique [51]. Cependant, l'introduction d'une telle structure nécessite l'étude de la stabilité du réseau dynamique et, dans tous les cas, l'apprentissage est nécessaire pour que le réseau de neurones soit fonctionnel. On rappelle que l'objectif de l'apprentissage est de déterminer les poids synaptiques du réseau qui sont modifiés selon des règles.

4.2.4. Apprentissage

Une des caractéristiques les plus intéressantes des réseaux de neurones est leur capacité à apprendre. L'apprentissage va permettre au réseau de modifier sa structure interne (poids) pour s'adapter à l'environnement, selon une règle d'apprentissage. L'apprentissage peut être *supervisé*, *non supervisé* ou *renforcé*.

Dans le premier cas, on connaît parfaitement la sortie désirée et on guide le réseau en lui apprenant à chaque étape le bon résultat. Ici, l'apprentissage consiste à comparer le résultat obtenu avec le résultat désiré, puis à ajuster les poids des connexions pour minimiser la différence entre la consigne et la valeur instantanée. L'apprentissage s'arrêtera dès que cette différence est inférieure à un seuil déterminé préalablement par le superviseur (fig.3).

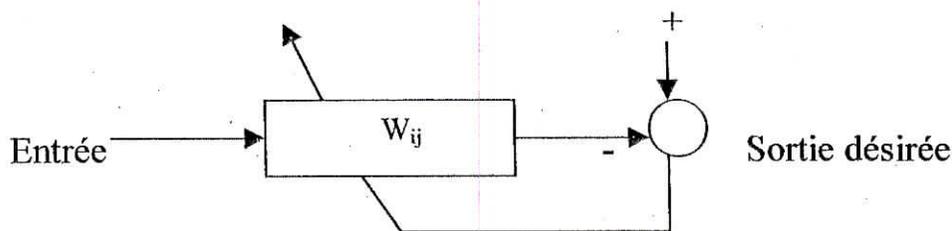


Fig.3 Apprentissage supervisé

Dans le cas de l'apprentissage non supervisé, encore appelé auto-organisation, l'apprentissage nécessite la présence d'entrées sans l'intervention d'un superviseur. Les sorties désirées ne sont pas présentées au réseau qui évolue librement jusqu'à un point d'équilibre. Ainsi les poids du réseau sont modifiés en fonction des critères internes comme la coactivation des données (fig.4).

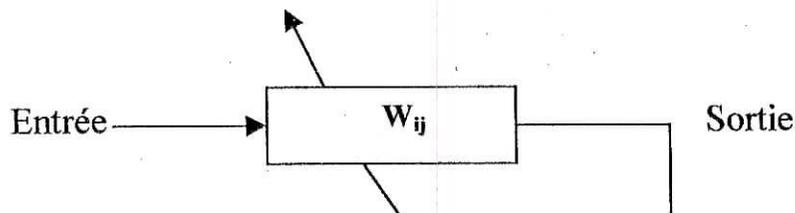


Fig.4 Apprentissage non supervisé

Quant à l'apprentissage *renforcé*, le réseau réajuste ses poids synaptiques suivant un critère de performance qui renforce ces poids suivant que la sortie délivrée est correcte ou non.

4.2.5. Réseaux à couches

Après avoir montré les limites du Perceptron en se basant sur le fait qu'un neurone ne peut pas représenter le *OU* exclusif devant les problèmes non linéaires, en 1969, Minsky et Papert's, ont résolu ce problème spécifique en introduisant une couche cachée qui donna naissance au Perceptron multicouches [52,53]. Dotés de non-linéarités, ces réseaux multicouches sont une généralisation du perceptron vers un réseau organisé en couches de neurones dont les entrées de chaque couche proviennent de la couche qui la

précède. Un réseau multicouche est donc constitué de n neurones d'entrées disposées en une seule couche, d'une ou plusieurs couches composées de p neurones dites couches cachées et une couche de sortie de m neurones. Notons que les neurones d'une même couche ne sont pas connectés entre eux, que chacune des couches reçoit des signaux de la couche précédente et transmet les résultats à la couche suivante. Les neurones d'une couche sont connectés à tous les neurones de la couche suivante par des connexions pondérées. Les réseaux multicouches utilisent l'algorithme de rétropropagation du gradient pour effectuer la mise à jour de la pondération du réseau. Dans la phase d'apprentissage par rétropropagation qui consiste en un entraînement du réseau, on présente, au réseau, des entrées. Une sortie réelle est alors calculée. Ce calcul est effectué, de proche en proche, de la couche d'entrée à la couche de sortie. C'est la phase de propagation avant ou relaxation du réseau. L'erreur entre les sorties désirées et les sorties réelles est rétropropagée dans le réseau vers l'arrière jusqu'à la couche d'entrée en passant par les couches cachées tout en ajustant les poids synaptiques du réseau. Le processus est répété en présentant successivement chaque exemple autant de fois que l'erreur reste supérieure à un certain seuil choisi (fig.5).

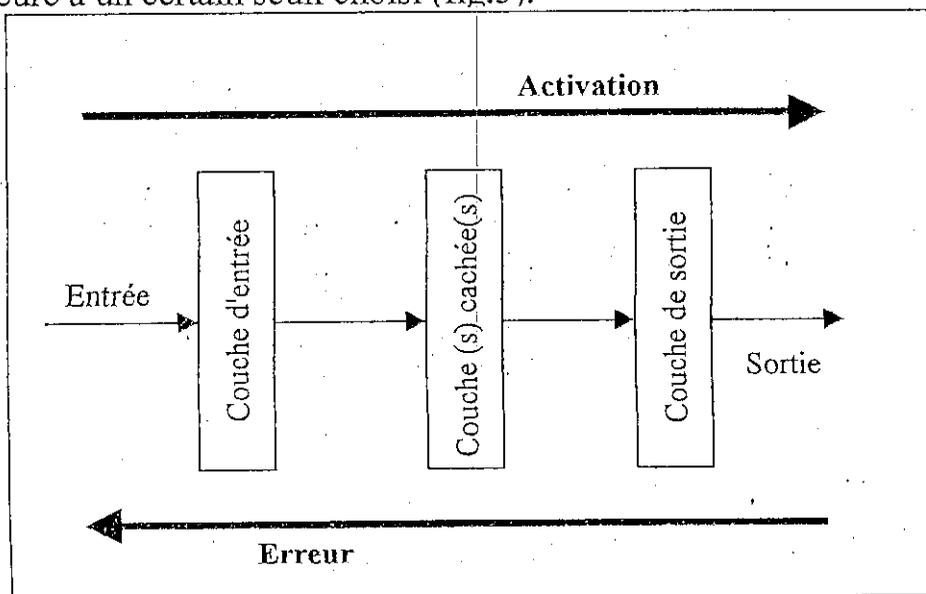


Fig.5 : Réseaux de Neurones multicouches :
algorithme de rétropropagation

4.2.6 Algorithme de rétropropagation

Le développement mathématique de la rétropropagation n'est, en principe, qu'une généralisation de Widrow-Hoff qui minimise la valeur moyenne de l'erreur quadratique $E(k)$ sur l'ensemble d'apprentissage et qui consiste à modifier les poids dans le sens inverse du gradient de l'erreur.

Posons tout d'abord:

$X=(X_1, X_2, \dots, X_n)$, vecteur des entrées

$Y=(Y_1, Y_2, \dots, Y_m)$, vecteurs des sorties désirées

$S=(S_1, S_2, \dots, S_m)$, vecteur des sorties effectivement obtenues

F , la fonction sigmoïde d'un neurone et F' sa dérivée,

O_j , la sortie du neurone j

I_i , l'entrée du neurone i ,

η , le pas du gradient à l'étape k

A l'étape k , l'erreur commise par rapport à chaque poids du réseau sur un passage complet peut se réécrire :

$$E(W) = \sum_k E^k(W) \quad (9)$$

où $E^k(W)$ est l'erreur commise à l'étape k telle que:

$$E^k(W) = \frac{1}{2}(S^k - Y^k)^2 = \frac{1}{2} \sum_{i=1, m} (S_i^k - Y_i^k)^2 \quad (10)$$

avec $Y_i^k = \sum W_{ij} x_j$, somme pondérée non seuillée.

L'apprentissage consiste à minimiser cette erreur quadratique commise sur l'ensemble des échantillons. On rappelle que cette erreur est considérée comme fonction des poids. La méthode consiste donc à utiliser

l'approximation d'une descente du gradient sur l'erreur. L'équation de la mise à jour des poids est donnée par :

$$\Delta W_{ij}(t) = -\eta \frac{\partial E}{\partial W_{ij}} \quad (11)$$

Avec η , taux d'apprentissage.

Pour tous les poids W_{ij} , il s'agit de calculer le gradient de l'erreur par rapport à chaque poids dans le réseau tel que :

$$\frac{\partial E^k}{\partial W_{ij}} = \frac{\partial E^k}{\partial I_i} \cdot \frac{\partial I_i}{\partial W_{ij}} \quad (12)$$

Or on a $\frac{\partial I_i}{\partial W_{ij}} = O_j$, puisque les sorties correspondantes sont indépendantes de W_{ij} .

En posant $\frac{\partial E^k}{\partial I_i} = d_i$, l'équation (12) devient:

$$\frac{\partial E^k}{\partial W_{ij}} = d_i O_j .$$

Ceci conduit à l'équation suivante:

$$\Delta W_{ij}(t) = -\eta \frac{\partial E}{\partial W_{ij}} = -\eta d_i O_j \quad (13)$$

Comme l'erreur quadratique des neurones de la couche de sortie est donnée par l'équation (10), le gradient de l'erreur s'écrit:

$$\frac{\partial E^k}{\partial I_i} = d_i = \frac{1}{2} \frac{\partial (\sum_j (S_i - Y_i)^2)}{\partial I_i} = (S_i - Y_i) \frac{\partial S_i}{\partial I_i} \quad (14)$$

car, seules les sorties S_i dépendent de I_i .

Comme $S_i = F(I_i)$, on a :

$$d_i = (S_i - Y_i) \cdot F'(I_i) \quad (15)$$

pour les neurones des couches cachées, on écrit :

$$d_i = \sum_h \frac{\partial E^k}{\partial I_h} \cdot \frac{\partial I_h}{\partial I_i} = \sum_h d_h \frac{\partial I_h}{\partial I_i} \quad (16)$$

où l'indice h porte sur les neurones sur lesquels le neurone i envoie des connexions. En effet, les entrées I_h des autres neurones ne dépendent pas de I_i . on a donc :

$$d_i = \sum_h d_h \cdot \frac{\partial I_h}{\partial O_i} \cdot \frac{\partial O_i}{\partial I_i} \quad (17)$$

or
$$\frac{\partial I_h}{\partial O_i} = \frac{\partial(\sum_p W_{hp} \cdot O_p)}{\partial O_i} = W_{hi} \quad (18)$$

et $O_i = F(I_i)$. On obtient finalement :

$$d_i = \sum_h d_h W_{hi} \cdot F'(I_i) \quad (19)$$

Ainsi, la règle de modification des poids à l'étape k est donnée par :

$$W_{ij}(k+1) = W_{ij}(k) - \eta d_i O_j \quad (20)$$

où l'on a :

$$d_i = (S_i - Y_i) \cdot F'(I_i) \quad \text{pour les neurones des couches de sortie et}$$

$$d_i = \sum_h d_h W_{hi} \cdot F'(I_i) \quad \text{pour les neurones des couches cachées où } h \text{ porte sur}$$

les neurones vers lesquels i envoie une connexion.

4.3. Mise au point d'un R.N.

Lors de la construction et l'entraînement du réseau, on doit tout d'abord procéder au choix d'une architecture du réseau car il n'existe pas de théorie de l'architecture pour déterminer le nombre de neurones de la couche cachée. On

doit aussi choisir les paramètres pour l'apprentissage ainsi que la fonction d'activation dont la plus utilisée est la sigmoïde $f(x) = \frac{1}{1 + e^{-\beta x}}$.

Le choix de la valeur du pas d'apprentissage η est difficile à obtenir et est généralement compris entre 0.1 et 0.01 car, trop petit, on augmente inutilement le temps d'apprentissage et lorsqu'il est trop grand, le processus d'apprentissage devient très rapide mais avec une mauvaise convergence vers la solution désirée (parasites d'oscillations). Plusieurs solutions ont été proposées pour remédier à cet inconvénient dont la plus utilisée est l'ajout d'un terme d'inertie α appelé "momentum" compris entre 0 et 1 à l'équation (11) qui contribue aussi effectivement à la réduction du temps d'apprentissage. Il représente l'importance à donner aux corrections précédentes des poids synaptiques qui se fait de la manière suivante:

$$W_{ij}(t+1) = W_{ij}(t) + \eta \Delta W_{ij}(t) + \alpha \Delta W_{ij}(t-1) \quad (21)$$

où les valeurs courantes de α sont choisies entre 0.6 et 0.9 [54].

La rétropropagation n'est qu'une méthode d'approximation de fonction semblable à celles employées en statistique. Elle est aussi une technique simple, robuste et efficace qui connaît aujourd'hui un grand intérêt auprès des chercheurs qui l'ont appliquée dans divers domaines.

Les bons résultats obtenus démontrent l'utilité pratique du réseau de neurones auquel est associée le perceptron multicouches [55].

4.4. Application des R.N. à la reconnaissance des graines

4.4.1. Principe

La méthode de reconnaissance des graines proposée est basée sur l'extraction de paramètres texturaux à partir de la matrice de cooccurrence et un réseau de neurones multicouches dont l'apprentissage est effectué par l'algorithme de rétropropagation. Les données d'application sont 280 images

de graines parmi les 400 précédemment citées à savoir l'avoine, le blé, les lentilles et l'orge avec des orientations et tailles différentes. Ces images sont, cette fois, saisies en niveaux de gris à l'aide de la caméra IonPC reliée à un micro-ordinateur via la carte d'acquisition d'images, la VideoMaker.

Deux groupes de graines ont été constitués. Le premier groupe est composé de 160 images dont 40 par classe seront utilisés pour l'apprentissage du réseau de neurones. Le second groupe est composé de 120 images, soit 30 par classe, seront utilisées pour la validation du réseau.

4.4.2. Prétraitement des images

Tout d'abord, pour ne conserver que l'information utile relative à la graine en l'extrayant du fond, l'opération *ET* logique est effectuée entre l'image en niveau de gris et l'image binarisée obtenue par l'application de la méthode de maximisation de la variance interclasse proposée par Otsu [56]. Cette méthode, malgré sa sensibilité aux effets d'ombre, a permis d'obtenir de bons résultats.

4.4.3. Extraction des paramètres de texture

La recherche de mesures permettant de quantifier l'information structurelle de la texture est fondamentale pour la caractérisation de la surface des graines. Ces mesures doivent être pertinentes et posséder un pouvoir discriminant afin de différencier les classes en présence. Plusieurs méthodes d'analyse de la texture ont été introduites dans la littérature [57]. La méthode statistique basée sur la dépendance spatiale des niveaux de gris demeure la plus utilisée [58]. On rappelle que cette méthode consiste à calculer une matrice de cooccurrence $M(i,j,d,\theta)$ dont chaque élément représente la

fréquence d'apparition d'un couple de pixels i, j séparés d'une distance d selon la direction θ prenant les quatre valeurs $\theta=0^\circ, \theta=\pi/2, \theta=\pi/4$ et $\theta=3\pi/4$ par rapport à l'horizontale. De cette matrice de cooccurrence de chaque type de graine, plusieurs paramètres de texture peuvent être calculés. Parmi ces paramètres, les plus importants sont le second moment angulaire (*SMA*) qui nous renseigne sur l'homogénéité de la texture, le contraste (*CONT*) qui mesure la variation locale de la texture et qui favorise les grandes transitions des niveaux de gris, l'entropie (*ENT*) qui évalue le degré d'organisation des pixels et le moment inverse (*MI*) dont la valeur devient de plus en plus faible à mesure que l'on s'éloigne de la diagonale principale et devient égale à l'unité s'il y a concentration de points autour de la diagonale principale. Pour une direction θ donnée, ces paramètres sont définis par les expressions suivantes :

$$Cont(\theta) = \sum_{i=0}^{N_g-1} \sum_{j=0}^{N_g-1} (i-j)^2 [M(i, j, d, \theta)]^2 \quad (22)$$

$$SMA(\theta) = \sum_{i=0}^{N_g-1} \sum_{j=0}^{N_g-1} [M(i, j, d, \theta)]^2 \quad (23)$$

$$ENT(\theta) = - \sum_{i=0}^{N_g-1} \sum_{j=0}^{N_g-1} M(i, j, d, \theta) \log M(i, j, d, \theta) \quad (24)$$

$$MI(\theta) = \sum_{i=0}^{N_g-1} \sum_{j=0}^{N_g-1} \frac{M(i, j, d, \theta)}{1 + (i + j)^2} \quad (25)$$

Dans notre application, les deux paramètres caractérisant le contraste et le second moment angulaire ont été calculés dans chacune des quatre directions

avec une distance 'd' égale à l'unité. Le tableau 1 montre un exemple de valeurs obtenues de ces paramètres qui diffèrent d'un cas à l'autre.

Graines	Direction	CONT	SMA
Avoine	0°	24.7219	0.0213
	45°	74.2825	0.0159
	90°	82.2954	0.0163
	135°	69.1197	0.0168
Blé	0°	44.4582	0.0287
	45°	89.4607	0.0233
	90°	97.4328	0.0256
	135°	84.3861	0.0254
Lentille	0°	41.2199	0.0650
	45°	45.1226	0.0564
	90°	66.5832	0.0569
	135°	70.1135	0.0570
Orge	0°	57.6817	0.0367
	45°	155.0185	0.0272
	90°	172.8518	0.0271
	135°	147.9945	0.0266

Tableau 1. Exemple de paramètres de texture des différentes graines

4.4.4. Apprentissage

Pour l'apprentissage du réseau de neurones, 160 images de graines ont été considérées. Pour chaque graine, les 2 paramètres définis précédemment dans les quatre directions sont calculés. Ainsi, la base d'apprentissage est constituée de 160 vecteurs prototypes à huit composantes. Pour classer ces différentes graines, on rappelle que plusieurs méthodes de classification supervisées ou non supervisées peuvent être utilisées. Mais dans notre cas, nous avons opté pour les méthodes neuronales. Nous avons utilisé un réseau de neurones à trois couches afin de réaliser un apprentissage (fig.6), avec une couche d'entrée composée de 8 cellules, une de sortie composée de 4 cellules et une couche cachée de 8 cellules.

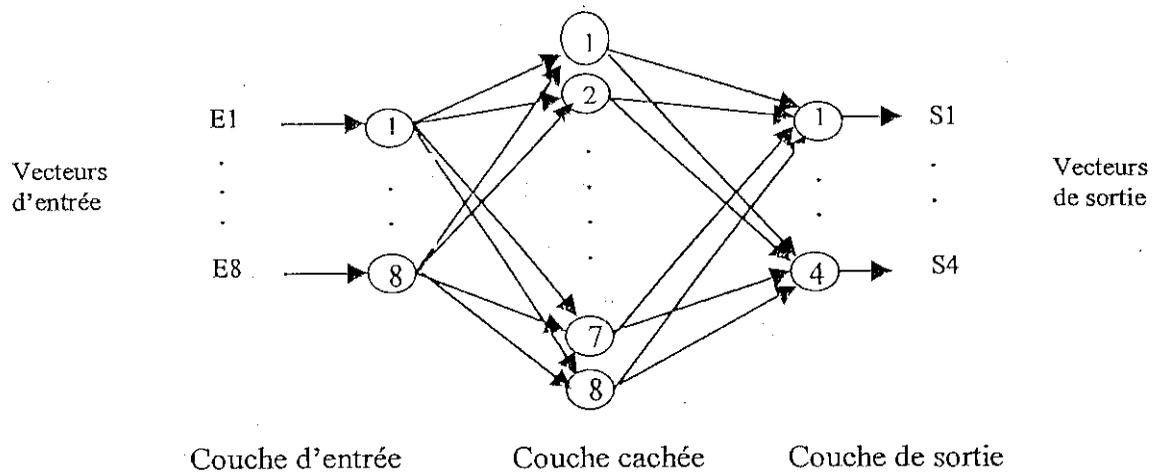


Fig.6 : Réseau de neurones multicouches

L'algorithme de rétropropagation utilisé pour l'entraînement du réseau consiste à minimiser l'Erreur Quadratique Moyenne :

$$E = \frac{1}{2} \sum_p \sum_n (S_n^p - d_n^p)^2 \quad (26)$$

où d_n^p représente la sortie désirée de la cellule n de la couche de sortie et p l'indice de prototype d'apprentissage qui, dans notre cas, est tel que :

$$0 < p \leq 160$$

La loi de modification des poids est donnée par la formule suivante :

$$W_{kj}(t+1) = W_{kj}(t) + \eta \Delta W_{kj}(t) + \alpha \Delta W_{kj}(t-1) \quad (27)$$

avec :

$$\Delta W_{kj}(t) = -\eta d_i O_j$$

$$d_i = (S_i - Y_i) \cdot F'(I_i) \quad \text{pour les neurones des couches de sortie et}$$

$$d_i = \sum_h d_h W_{hi} \cdot F'(I_i) \quad \text{pour les neurones des couches cachées}$$

Le paramètre α est compris entre les valeurs 0,6 et 0,9. Quant à η , il est compris entre 0,01 et 0,1. F' est la dérivée de la fonction d'activation F .

4.5. Tests et Résultats

Pour réaliser l'apprentissage, l'erreur globale a été fixée à 0,001. Le nombre d'itérations obtenu est de 3053. Ainsi, toutes les 160 graines ont été bien classées, ce qui donne un taux de reconnaissance de 100%. Afin de valider les résultats obtenus lors de l'apprentissage, nous avons considéré le deuxième groupe de 30 autres graines de chacune des quatre classes. Après le calcul des deux paramètres dans les quatre directions pour chacune de ces graines, le réseau de neurones affecte chacune de ces graines à la classe correspondante parmi les quatre. La cellule dont l'activation est maximale nous donne l'espèce de graine. Ainsi, sur l'ensemble de ces graines, douze ont été mal identifiées conduisant ainsi à un taux de reconnaissance de 90%. Il s'agit de sept graines de blé confondues avec l'avoine, de trois graines de lentille classées comme des graines de blé et de deux graines d'orge classées comme des graines d'avoine.

4.6. Conclusion

Dans ce chapitre, nous avons montré que l'approche de la reconnaissance des formes utilisant les paramètres de texture à l'aide d'un réseau de neurones fournit des résultats très intéressants et a réussi là où l'approche utilisant les paramètres de forme a échoué [59]. La combinaison des deux méthodes peut s'avérer intéressante car chacune peut combler les lacunes de l'autre. Par ailleurs, le choix d'autres paramètres de texture et d'autres méthodes d'analyse de texture sont à expérimenter tout comme l'utilisation d'un nombre plus important de graines ou d'autres espèces de graines. Il est également intéressant d'introduire une classe de rejet afin de rejeter les impuretés et tout corps étranger par rapport aux graines étudiées.

Conclusion Générale

Conclusion Générale

Le travail effectué dans ce mémoire entre dans le cadre d'une application de la reconnaissance des formes au problème de tri et de sélection automatique d'objets.

L'application développée dans ce mémoire concerne le tri de graines de semence dont la particularité réside dans la variabilité d'une même espèce nécessitant de par cela un choix judicieux des paramètres qui sont de type morphologique, de couleur ou de texture. Cette application peut s'étendre à n'importe quel autre type d'objets puisqu'on retrouve toujours le même schéma de base composé des phases de paramétrisation, d'apprentissage et de reconnaissance ou de décision. Au cours de la phase d'apprentissage ou de classification automatique, on peut se retrouver devant différentes situations selon la disposition ou non de l'information *a priori*. Dans notre cas, si les données sont bien connues par l'utilisateur, le principe d'une classification automatique non paramétrique et non supervisée a été adopté. C'est ainsi, après la normalisation et la réduction des données, l'agrégation autour des centres mobiles combinée avec l'ACP a été utilisée.

Les quatre classes correspondant aux graines de blé, d'orge, d'avoine et de lentille ont été formées dont deux, celle de l'orge et de l'avoine se chevauchent et ce, quel que soit le type de paramètres utilisés, morphologiques ou de texture. D'ailleurs, l'extraction de ces paramètres dépendent beaucoup de la qualité des images et nécessitent souvent des traitements préalables de type binarisation, suivi de contour et filtrage des images. Ces opérations sont parfois indispensables pour réussir l'opération de reconnaissance, surtout dans le cas particulier où l'éclairage est mal adapté.

Dans notre application, deux approches, l'une basée sur les paramètres morphologiques et l'autre sur des paramètres de texture, ont été donc utilisées. Avec la première, après la formation des classes, la décision est basée sur la détermination de l'erreur quadratique moyenne entre signatures polaires moyennes des différentes classes et de celle de l'objet inconnu à reconnaître. Le taux de reconnaissance de 100% pour la classe de blé et de lentille a été obtenu. Par contre, du fait de leur ressemblance, les classes orge et avoine se chevauchent avec un taux de confusion voisinant les 7,5%. Ce taux serait, sans doute, amélioré en utilisant un matériel d'acquisition plus performant.

Avec l'approche utilisant les paramètres de texture à l'aide d'un réseau de neurones des résultats très intéressants ont été obtenus. En effet, la validation des résultats a montré qu'un taux de reconnaissance de 90% a été obtenu. Quelques graines de blé ont été confondues avec l'avoine et peu de graines de lentille et d'orge ont été confondues respectivement avec celles de blé et d'avoine.

Cependant, d'autres méthodes utilisant les paramètres de couleur telles que la teinte, la saturation et la luminance peuvent aussi être expérimentées dans ce type particulier d'application.

Partant du fait que l'approche utilisant les paramètres morphologiques a échoué là où l'approche utilisant les paramètres de texture à l'aide d'un réseau de neurones a réussi et inversement, une combinaison entre les trois méthodes pourrait fournir des résultats meilleurs si l'on pourra trouver un moyen intelligent pour combiner les résultats fournis par chaque méthode.

Par ailleurs, le choix d'autres paramètres de texture et d'autres méthodes d'analyse de texture sont à expérimenter tout comme l'utilisation d'un nombre plus important de graines ou d'autres espèces de graines. Il serait également intéressant d'introduire une classe de rejet afin de rejeter les impuretés et tout corps étranger par rapport aux graines étudiées.

Bibliographie

BIBLIOGRAPHIE

- [1]-R.O. DUDA and P.E. HART, **Pattern Classification and Scene Analysis**, Wiley, 1973
- [2]-A BELAID et Y. BELAID, **Reconnaissance des Formes: Méthodes et Applications**, InterEditions, Paris, 1992
- [3]-J.C. SIMON, **La Reconnaissance des Formes par Algorithmes**, Masson, 1984
- [4]-E. IAGOLNITZER, **Présentation Des Programmes d'Analyse en Composantes Principales**, MLIF, Informatique et Sc. Hum., n°. 26, 1975
- [5]-J.P. BENZEKRI, **L'Analyse des Données, La Taxinomie**, Dunod, 1980
- [6]-A. TOUZANI, **Classification Automatique par Détection des Contours des Modes des Fonctions de Densité de Probabilité Multivariées et Etiquetage Probabiliste**, Thèse de Docteur ès sciences Physiques, UST Lille-F-Artois, 1987
- [7]-J.T. Tou, R.C. Gonzales, **Pattern Recognition Principles**, Addison Wesley, Reading Massachussets, 1974
- [8]-K.S. Fu, **Digital Pattern Recognition, Communication and cybernetics**, , vol.10, New-york, Springer, 1976
- [9]-J.C.H. Chen, **Statistical Pattern Recognition**, New York, Hayden, 1973
- [10]-J.G. POSTAIRE, **De l'Image à la Décision**, Dunod, 1987

- [11]-E. DIDAY, A. SCHROEDER and Y. OK, **The Dynamic Cluster Method in Pattern Recognition**, Proceeding IFIP-1974, 691-7., 1974
- [12]-A. TOUZANI et J.G. POSTAIRE, **Clustering by Mode Boundary Detection**, IEEE Trans. Patt. Anal. Machine Intell., 1987
- [13]-B.S. EVERITT, **Cluster Analysis**, John Wiley & Sons, New York, 1974
- [14]-E. PARZEN, **On Estimation of Probability Density Function and Mode**, Ann. Math. Stat., Vol. 33, pp. 1065-1076, 1962
- [15]-J. KITTLER, **A Method For Determining k Nearest Neighbors**, Kibernet, vol. 7, 313-315, 1978
- [16]-M. DIAF, S. DJENNOUNE et S. SCENALOWICZ, **Estimateur des Plus Proches Voisins et Filtre Médian en Classification Automatique par Les K Plus Proches Voisins**, 1-ier Congrès National de Mathématiques, Tizi-Ouzou, Nov. 1994
- [17]-M. BRUYNOOGHE M., **Classification Ascendante Hiérarchique de Grands Ensembles de Données. Un Algorithme Fondé Sur La Construction de Voisinages réductibles**, Cahiers de l'Analyse des Données, vol. 3, n°. 1, 35-46, 1978
- [18]-J.R. ULLMAN, **Pattern Recognition Techniques**, Butterworths, Londres, 1973
- [19]-M. DIAF and K.K. CHOPRA, **A Mixed Pattern Classification Method**, Studies In Informatics and Control, Vol. 3, n°. 1, pp. 33-42, 1994

- [20]- R. Murrieta Cid et al., **Color segmentation in principal regions for natural outdoor scenes**. 3rd workshop on Electronic Control and Measuring Systems, Toulouse, 1997
- [21]- C.Duperthuy et J.M Jolion, **Simplification d'images couleurs par analyse de contraste en multirésolution**, 16^{ième} colloque CRETSI, Grenoble, 1997
- [22]- A. Rosenfeld and al., **Digital Picture Processing**, 2nd Edition, vol.2, Academic Press, 1982
- [23]- P. Lassere, **Vision pour la robotique en environnement naturel**. Thèse de l'université Paul Sabatier Toulouse, France, LAAS, 1996
- [24]-Y. Ohta, **Knowledge-Based Interpretation of Outdoor Natural Color Scenes**. Morgan Kaufman Publishers, INC., Palo Alto, 1985
- [25]-F. ADE, **Characterization of Texture by Eigenfilters**, Signal Processing, 1983
- [26]-B. JULESZ, **Experiments In The Visual Perception of Texture That Agree In Second Order Statistics**, Perception, pp. 391-405, 1975
- [27]- S. WANG-CHEUNG LAM, **Texture Feature Extraction Using Gray Level Gradient-Based Co-occurrences Matrices**, IEEE trans. on Pattern Analysis and Machine Intelligence, p.267-271, 1996
- [28]-J-P. COCQUEZ et SYLVIE PHILIPP, **Analyse d'images, filtrage et segmentation**, Masson, 1995
- [29]-M. DIAF et M. DJEDDI, **Effet du Filtrage de Fourier Sur les Paramètres de Texture des Images de Télédétection**, Bulletin de L'Office de la Recherche en Géologique et Minière, 1994

- [30]-F. DERAVID and S.K. PAL, **Gray Level Threshold Using Second-Order Statistics**, Pattern Recognition Letter 1, pp. 417-422, 1983
- [31]- R. HUMMEL, **Image Enhancement by histogram transformation**, CGIP, vol. 6,184-195, 1977
- [32]- CHEHDI K. AND COQUIN D.. **Binarisation d'images par seuillage local optimal maximisant un critère d'homogénéité**. Troisième colloque GRETSI Juan -les- pins, France,1991
- [33]- P.K. SAHOO, S. SOLTANI, A.K.C. WONG AND Y.C. CHEN, **A Survey of Thresholding Techniques**, Computer Vision Graphics and Image Processing, pp. 233-260, 1988.
- [34]-M. DIAF and L. ZEBROWSKI, **Segmentation d'Images Par Etiquetage Probabiliste**, JTEA'91 (Journées Tunisiennes d'Electrotechnique et d'Automatique), Tunisie, pp. In 4.1-4.4, 1991
- [35]-S. PELEG and A. ROSENFELD, **Determining Compatibility Coefficients For Curve Enhancement Relaxation Process**, IEEE Trans. on Systems, Man. and Cybernet. vol. SMC-8, n°. 7, pp. 548-55, 1978
- [36]-S.W. ZUKER, **Region Growing : Childhood and Adolescence**, Comput. Graphics Image Processing 5, pp. 382-399, 1976,
- [37]-F. HARROIS-MONIN, **Le Marché des Images de la Terre**, Sc.& Vie 820, 88-103, 1986
- [38]-A.J. DANKER and A. ROSENFELD, **Blob Detection By Relaxation**, IEEE Trans. On Pattern Anal. and Mach.Intell., vol. PAMI-3, n°. 1, pp. 79-92, 1981

- [39]-M. BASSEVILLE, **Detection de Contours: Methodes et Etudes Comparatives**, Annales des Télécommunications, 34, n°11-12, 1979.
- [40]-A. MARTELLI, **Edge Detection Using Heuristic Search Methods**, Comput. Graphics image Processing 1, pp. 169-182, 1972
- [41]-Mac Culloch & al., **A logical calculus of the ideas immanent in nervous activity**, bulletin of Mathematical biophysics, 1943
- [42]- F. ROSENBLATT, **The perceptron : a probabilistic model for information storage and organization in the brain**, Psychologie Review, 1965
- [43]- D.O. HEBB, **The organization of the behaviour**, J. Wiley & Sons, 1949
- [44]- D.E. RUMMELHART & al., **Learning internal representation by error propagation in parallel distributed processing : explorations in the microstructures of cognition**, Mitt Press, Cambridge; 1986
- [45] - J. F. JODOUIN, **Les réseaux neurométriques**, Hermes, 1994.
- [46]-B. MULLER AND AL. **Neural Networks**, Spring Verlag 1995
- [47]-X. LUO & al., **Real over transfer capability calculations systems using multi-layer feedforward neural networks**, IEEE transactions on Neural Networks, vol.15, n° 2, p.903-908,2000
- [48]-G. & al. WIDROW , **Adaptative swintching circuits**, Institue of Radio Engineers, Western Eletronic Show and Convention Record, 1960

- [49]- BEN J.A. KRÖSE AN P. PATRICK VAN DER SMAGT, **An introduction to Neural Network**, The university of Amsterdam , fifth edition ,1993
- [50]- J.A. FELDMAN & D.H. BALLARD, **Connectionist models and their properties**, Cognitive Science 6, p.205-254, 1982
- [51]- G. DREYFUS, Y. IDAN, **The canonical form of non-linear discrete-time models**, Neural Computation vol. 10, n°1, 1998
- [52]-M. MINSKY & A. PAPERT, **Perceptrons : An introduction to computational Geometrical** , MIT Press, 1969
- [53]-Q.SONG, **Robust algorithm of multilayer neural networks for identification of nonlinear dynamic system**, IEE proc. Contreol theory Appl., vol 1, 1998
- [54]-R.P. LIPPMANN, **An introduction to computing with neural nets**, IEEE ASSP mag., p4-22, april 1987
- [55]- **F. DIAF**, M. GUERTI, K. HAMMOUCHE ET M. DIAF, **Contribution au tri automatique de graines de semences**, Sidi-Bélabès, 1999.
- [56]- L.G. SHAPIRO, **Data Structures For Picture Processing: A Survey** Computer Graphics Image Processing 11, pp. 162-184, 1979
- [57]- R.M. HARRALICK, K. SHANMUGAM AND I. DINSTEIN, **Textural Features for image classification**, IEEE trans. On systems man. And cybernetics, vol.6, pp. 610-621, 1973.
- [58]-C. CHEN, R.C. Dubes, **Discrete MRF Model parameters as features for texture classification**, Proc. IEEE on Syst. Man.Cybernet. p. 1-6, 1990

[59]-F. DIAF& al., **Automatic Identification And Counting Graines by Image Analysis**, Proceedings Intern. A.M.S.E., vol.2, 1993.