

UNIVERSITE KASDI MARBAH OUARGLA
Faculté des Nouvelles Technologies de l'information et de La Communication
Département d'Informatique Et des Technologies de L'information



Mémoire

MASTER ACADEMIQUE

Domaine : Mathématique Et Informatique.

Filière : Informatique Académique.

Spécialité : Informatique Fondamental

Présenté par :

✓ Maou Hadia

Thème

***classification des feuilles de plantes avec la carte
de KOHONEN***

Soutenu publiquement le :

10/06/2015

Devant le jury :

M.	Bachir Said	Président	UKM Ouargla
M.	Bekkari Fouad	Encadreur/rapporteur	UKM Ouargla
M.	Korichi Meriem	Examineur	UKM Ouargla

Année Universitaire : 2014 /2015

Remerciements

Je remercie tout d'abord notre Dieu qui m'a donné la force et la volonté pour élaborer ce travail.

J'adresse mes vifs remerciements à mon encadreur

Mr. Bekkari Fouad

qui m'a aidé durant mon travail et par sa patience et ses précieux conseils dont Il m'a entouré.

Sans son aide, mon travail n'aurait pas vu la lumière.

Je remercie vivement les membres du jury qui m'ont fait l'honneur d'accepter de juger notre travail.

ma reconnaissance aussi à tous ceux qui ont collaboré à ma formation en particulier les enseignants du département d'Informatique, de l'université KASDI MARBAH OUARGLA.

*Aussi à mes collègues de la promotion 2014-2015
Master Informatique,*

Je remercie également tous ceux qui ont participé de près ou de loin à élaborer ce travail.

Dédicaces

A

mon père et ma mère

mes frères et mes sœurs

mes très proche amies: Imane .B

tous mes amis surtout : Khadija.K

.Wafa.G. Imane.G

toute ma famille

tous...

Hadia

Table des matières

Table des matières	I
Liste des figures	III
Liste des tableaux.....	IV
Introduction générale	1

Chapitre I:les descripteurs des images

I.1.Introduction.....	2
I.2.Notions de base.....	2
I.2.1.Les images numériques.....	2
I.2.2.Les types d'images.....	3
I.2.3.Les formats d'image.....	4
I.2.4.Les descripteurs d'image.....	5
I.3.Conclusion.....	13

Chapitre II:les cartes auto-organisatrice

II.1.Introduction.....	16
II.2.Réseaux de de neurones.....	16
II.3.Cartes auto-organisatrices de Kohonen.....	16
II.3.1.Architecture des cartes de Kohonen.....	17
II.3.2.La règle d'apprentissage.....	19
II.3.3.Algorithme d'apprentissage d'une carte de Kohonen.....	20
II.3.4.Domaines d'application.....	21
II.3.5.Les avantages de cartes de Kohonen.....	22
II.3.6.Inconvénients des cartes de Kohonen.....	22
II.4.Conclusion.....	22

Chapitre III:La conception

III.1.Introduction.....	24
III.2.Présentation de langage UML.....	24

III.3.La conception de l'application.....	24
III.4.Les diagrammes utilisés.....	26
III.4.1.Le diagramme des cas d'utilisation.....	26
III.4.2.Diagramme de séquences.....	26
III.4.3.Le diagramme de classes.....	32
III.5.Conclusion.....	32

Chapitre IV: La réalisation

IV.1.Introduction.....	34
IV.2.les outils de développement.....	34
IV.2.1.Environment de travail.....	34
IV.2.2.Le langage de programmation.....	34
IV.2.3.Présentation XML.....	35
IV.3.Description de l'application.....	35
IV.4.Test et Résultats.....	40
IV.5.Conclusion.....	40
Conclusion générale.....	41
Bibliographie	42
Glossaire	45

Liste des figures

Fig I.1: les types d'images.....	3
Fig I.2: Les trois primaires additives de lumière(Rouge, Vert, et Bleu).	6
Fig I.3: Une image et son histogramme: on distingue clairement le pic lié aux blancs.	7
Fig II.1: Des neurones voisins sur la carte représentent des objets assez "proche" dans l'espace des données d'entrées	17
Fig II.2: Structure d'une carte auto-organisatrice.	18
Fig II.3: Les voisinage les plus utilisées des cartes de kohonen: (a) rectangulaire (b) hexagona.	19
Fig III.1: Architecture conceptuelle du système.....	25
Fig III.2: Diagramme Cas d'utilisation.....	26
Fig III.3: cas d'utilisation «classification».....	27
Fig III.4: diagramme de séquences «classification».....	27
Fig III.5: cas d'utilisation «calcul moment Hu»	28
Fig III.6: diagramme de séquences «calcul moment Hu»	28
Fig III.7: cas d'utilisation «Mise à jour».....	28
Fig III.8: diagramme de séquences «ajout un nouveau type»	29
Fig III.9: diagramme de séquences «ajout un nouvelle base».....	30
Fig III.10: cas d'utilisation «Initialisation».....	31
Fig III.11: diagramme de séquences «initialisation»	31
Fig III.12: Diagramme de classe.....	32
Fig IV.1: L'interface de notre application.....	35
Fig IV.2: La fenêtre classification.	36
Fig IV.3: interface avec le Menu.	36
Fig IV.4: Charge un image.	37
Fig IV.5: Interface de résultats de Classification.....	37
Fig IV.6: la résultat de calcule moment Hu.....	38
Fig IV.7: les interfaces initialisation.....	39
Fig IV.8: les interfaces ajout nouveau type.	39
Fig IV. 9: l'interface d'item Aide.	40

Liste des tableaux

Tabl I.1: les principales caractéristiques des formats5

Tabl IV.1: Résultats de test.....40

Introduction générale

L'image est généralement l'un des moyens les plus importants qu'utilise l'homme pour communiquer avec son entourage. C'est un moyen de communication universel dont, la richesse du contenu permet aux êtres humains de tout âge et de toute culture de se comprendre. C'est aussi le moyen le plus efficace pour communiquer, chacun peut analyser l'image à sa manière, pour en dégager une impression et d'en extraire des informations précises.

la classification est l'une des la plus importante et la plus courante tâche dans le domaine imagerie en même temps, elle est l'un des problèmes majeurs rencontrés[HIL, 09]. De nombreuses approches de classification sont proposées dans la littérature, le plus souvent basées sur le calcul d'invariants utilisés pour alimenter la classification.

Notre travail s'inscrit au domaine de la classification des images, on va réaliser une application pour la classification des feuilles de plantes. En utilisant la carte auto-organisatrice de KOHONEN comme méthode de classification et on utilise les moments invariants de Hu comme entrées pour faire l'apprentissage.

Pour réaliser notre application, nous avons organisé le mémoire en quatre chapitres comme suit :

Le premier chapitre présente quelque notion de base sur l'image numérique et quelque descripteur globale de l'image.

Le second chapitre présente la méthode choisie, ainsi que les différents étapes qui la constituent.

Dans le troisième chapitre, nous allons faire la conception de notre application en utilisant le langage UML, pour faciliter l'implémentation.

Le quatrième chapitre concrétise notre application de classification des feuilles des plantes à l'aide de langage Java.

En conclusion nous présentons les points de ce travail et quelques perspectives.

Chapitre I

les descripteurs des images

L'image est la représentation d'une scène acquise à l'aide de systèmes de production d'images (appareils photographique, caméra, radiographies, scanner, sonar,.....). Dans ce chapitre, je présente les différentes technologies derrière cette notion.

I.1. Introduction

Les images nous semblent parfois plus faciles à comprendre que des textes. Afin de permettre une lecture plus consciente de ce que l'image véhicule, Il faut analyser cette image. L'analyse d'images est la reconnaissance des éléments contenus dans l'image. Cette notion trouve sa place dans de nombreux domaines tels que les sciences de la vie, la robotique et encore la médecine .

Le but de l'analyse d'images réside dans la description (forme, couleur, texture) et/ou la quantification (nombre, densité) des divers objets qui composent l'image étudiée.

Dans ce chapitre, nous allons présenter une brève définition d'image numérique et quelques notions de base tels que les types des images, les formats et différents types de descripteurs.

I.2. Notions de base

I.2.1. Les images numériques

Une image est la représentation d'une scène acquise à l'aide de systèmes de production d'images (appareils photographique, caméra, radiographies, scanner, sonar,.....). Sa forme peut être analogique (ex: photographie, vidéo..) ou numérique [JAC, 01].

Le terme d'image numérique désigne, dans son sens le plus général, toute image qui a été acquise, traitée et sauvegardée sous une forme codée représentable par des nombres (valeurs numérique).

La numérisation est le processus qui permet de passer de l'état d'image physique (image optique par exemple) qui est caractérisée par l'aspect continu du signal qu'elle représente (une infinité de valeur dans l'intensité lumineuse par exemple), à l'état d'image numérique qui est caractérisée par l'aspect discret (l'intensité lumineuse ne peut prendre que des valeurs quantifiées en un nombre fini de points distincts).

Cette forme numérique permet une exploitation ultérieure par des outils logiciels sur ordinateur [Web, 01].

I.2.2. Les types d'images

Il existe deux grandes familles d'images numériques matricielle et vectorielle :

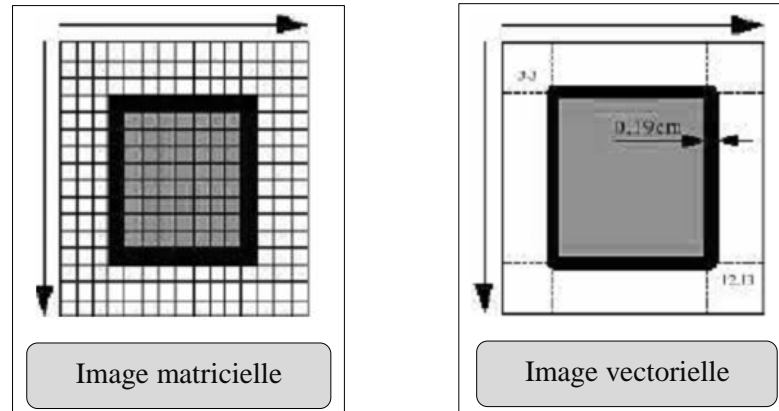


Fig I.1: les types d'images

a) Image Matricielle

Il s'agit d'images pixellisées⁽¹⁾ (appelées aussi images raster ou bitmap), Elle est composée comme son nom l'indique d'une matrice (tableau) de points, appelés pixels (voir fig I.1), chacun de ces points possédant une ou plusieurs valeurs décrivant sa couleur.

Avec ce type d'image il est possible de représenter des scènes complexes (photographies), mais avec une perte de qualité si on agrandit trop l'image [MOU, 07].

b) L'image vectorielle

Une image vectorielle en informatique, est une image numérique composée d'objets géométriques individuels (segments de droite, polygones, arcs de cercle, etc.) définis chacun par divers attributs de forme, de position, de couleur, etc. voir (fig I.1). [LAA, 14].

Ce type d'image permet de représenter des scènes simples et leur faire subir des transformations (agrandissement, rotation) sans perte de qualité [MOU, 07].

1- Le **pixel** (souvent abrégé px ou p) est l'unité de base permettant de mesurer la définition d'une image numérique matricielle.

I.2.3. Les formats d' image

Un format d'image est une représentation informatique de l'image. Le tableau suivant récapitule brièvement les principales caractéristiques des formats les plus courants.

Format	Type & Statut	Description	Utilisation
GIF Graphics Interchange Format	matriciel format propriétaire (CompuServe - Unisys)	<ul style="list-style-type: none"> • Supporté par tous les navigateurs et reconnu par tous les logiciels de traitement d'image. • 256 couleurs • algorithme de compression sans perte d'information • affichage progressif possible (GIF entrelacé) • gestion de la transparence • mono-couche 	<ul style="list-style-type: none"> • web • graphiques simples (boutons, icônes, schémas)
TIFF Tagged Image File Format	matriciel standard	<ul style="list-style-type: none"> • un des premiers formats graphiques . • reconnu par tous les logiciels de traitement d'image. • non supporté par les navigateurs • image de haute qualité (haute résolution et jusqu'à 16,7 millions de couleurs) • compression sans pert • mono-couche • format très performant, mais lourd 	<ul style="list-style-type: none"> • format de stockage pour images scannées • photographie
JPEG Joint Photographics Expert Group	matriciel norme ISO	<ul style="list-style-type: none"> • compression à qualité contrôlé (avec perte) • jusqu'à 16,7 millions de couleurs • format extrêmement léger • mono-couche 	<ul style="list-style-type: none"> • web • format de diffusion pour images réelles
PNG Portable Network Graphics	matriciel standard W3C	<ul style="list-style-type: none"> • format émergeant (objectif principal : remplacer GIF par un format plus performant et libre) • affichage progressif • compression sans perte (taux de compression élevé) • correction gamma et chromacité (color management system), <i>i.e.</i> adaptation des couleurs de l'image à l'écran • multi-couche 	<ul style="list-style-type: none"> • web • de l'image simple à la photographie

SVG Scalable Vector Graphics	mixte standard W3C	<ul style="list-style-type: none"> • format vectoriel défini par une application (DTD) XML • permet d'inclure des images matricielles • fonctionnalités de lien avec l'utilisation de Xlink et Xpointer 	<ul style="list-style-type: none"> • web • graphiques "simples"
CGM Computer Graphics Metafile	mixte normes ISO	<ul style="list-style-type: none"> • format vectoriel récemment élargi à l'inclusion d'images matricielles • format structuré • multi-couche • métadonnées incluses dans l'image 	<ul style="list-style-type: none"> • web • illustrations techniques (géophysique, aéronautique..)

Tabl I.1: les principales caractéristiques des formats

I.2.4. Les descripteurs d'image

La littérature dans ce domaine étant très riches, mais nous allons dans ce qui suit présenter les méthodes les plus importantes et les plus pertinentes qui sont de trois types:

- ↪ La description globale de l'image.
- ↪ La description locale de l'image .
- ↪ Les descripteurs spécifiques.

4.1. Le description globale de l'image

La description globale d'une image est l'extraction des caractéristiques symboliques comme la couleur, la texture, la forme...etc. de la totalité d'image .

4.1.1. Descripteurs de la couleur

La couleur est une caractéristique riche d'information et très utilisée pour la représentation des images. Elle forme une partie significative de la vision humaine [BOU, 09]. L'utilisation de la couleur pour l'extraction des primitives repose sur deux principaux choix: l'espace colorimétrique et le mode de représentation de la couleur dans cet espace [BED, 11].

A. Les espaces de couleurs

Une couleur est généralement représentée par trois composantes. Ces composantes définissent un espace de couleurs. Plusieurs études ont été réalisées sur l'identification d'espaces colorimétriques le plus discriminants mais sans succès car il n'existe pas d'espace de couleurs idéal [BED, 11].

L'espace RGB est l'espace de couleur le plus utilisé pour la représentation de la couleur. Il est composé des trois couleurs primaires: rouge, vert et bleu. Ce modèle est aussi le plus utilisé pour reproduction de la couleur sur les dispositifs d'affichage tels que la télévision et les écrans des ordinateurs. Ces trois couleurs sont appelées les couleurs primaires additives (Fig I.2). Par la variation de leur combinaison, d'autres couleurs peuvent être obtenues. Ce modèle peut être représenté sous forme d'un cube unitaire avec : noir (0,0,0), blanc(1,1,1), rouge (1,0,0), vert (0,1,0), et le bleu (0,0,1). Et les couleurs secondaires: cyan (0, 1,1), magenta (1, 0,1), et le jaune (1, 1,0) [MES, 09].

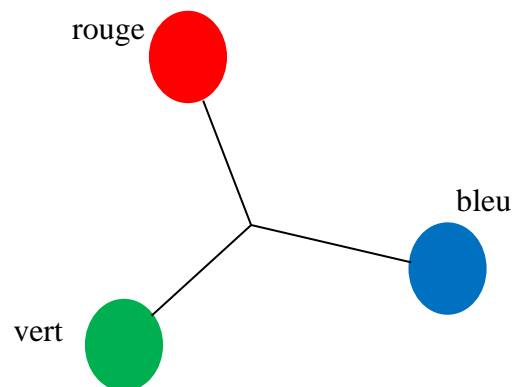


Fig I.2: Les trois primaires additives de lumière(Rouge, Vert, et Bleu).

B. L'histogramme

Le descripteurs de couleurs le plus courant et le plus rencontré dans littérature est l'histogramme de couleurs. Il est défini comme une fonction discrète qui associe à chaque valeur d'intensité le nombre de pixels prenant cette valeur. La détermination de l'histogramme est donc réalisée en comptant le nombre de pixel pour chaque intensité de l'image [Web, 02]. Le figure I.3 représente une image et son histogramme.

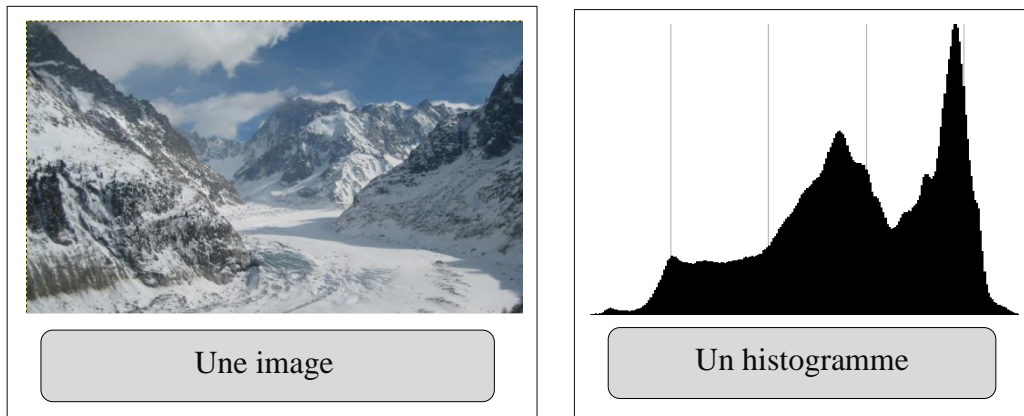


Fig I.3: Une image et son histogramme: on distingue clairement le pic lié aux blancs.

4.1.2. Descripteurs de texture

La texture est définie comme la répétition d'un motif créant une image visuellement homogène. Plus précisément, la texture peut être vue comme un ensemble de pixels spatialement agencés selon un certain nombre de relations spatiales, ainsi créant une région homogène. De ces définitions, les recherches sur la modélisation des textures se sont portées sur la caractérisation de ces relations spatiales [GUE, 07].

De nombreuses approches et modèles ont été proposées pour la caractérisation de la texture. Parmi les plus connues, on peut citer : les méthodes statistiques, les méthodes géométriques et les méthodes fréquentielles [JIA, 01].

A. Les méthodes statistiques

Ce sont les méthodes basées sur des évaluations quantitatives de la distribution de niveaux de gris⁽¹⁾. Elles étudient les relations entre un pixel et ses voisins. Elles sont utilisées pour caractériser des structures fines, sans régularité apparente [BED, 11].

B. Les méthodes géométriques

Ce type de méthodes tient compte de l'information structurelle et contextuelle de l'image. Ces méthodes sont particulièrement bien adaptées aux textures macroscopiques (structurelles). En effet, la description de la texture est faite par une extraction explicite de primitives (primitives: ensemble connexe de pixels qui partagent des propriétés similaires) et

1- **le niveau de gris**: représente la luminosité d'un pixel, lorsque valeurs de ses composantes de couleur sont identiques.

des règles de placement de ces primitives, par le biais d'attributs appelés attributs géométriques [BAB, 11].

C. Les méthodes fréquentielles

L'une des méthodes de description de la texture les plus utilisées concerne les propriétés fréquentielles et s'appuie sur la transformée de Fourier, le filtre, les ondelettes etc. Elle repose sur l'analyse d'une fonction de densité spectrale dans un domaine fréquentiel [BED, 11].

4.1.3. Descripteurs forme

La forme est l'un des attributs bas niveau également le plus utilisé pour décrire le contenu visuel des images [BEL, 08]. De nombreuses solutions ont été proposées pour représenter une forme, nous distinguons deux catégories de descripteurs de formes [BED, 11].

- ↪ **les descripteurs basés sur les régions:** qui décrivent les objets selon la distribution spatiale des pixels qui les constituent.
- ↪ **Les descripteurs basés sur les frontières (contours):** qui décrivent les objets selon leur contour externe [BEL, 08].

Les premiers font classiquement référence aux moments invariants et sont utilisés pour caractériser l'intégralité de la forme d'une région. Ces attributs sont robustes aux transformations géométriques comme la translation, la rotation et le changement d'échelle. La seconde approche fait classiquement référence aux descripteurs de Fourier⁽¹⁾ et porte sur une caractérisation des contours de la forme. Nous présentons dans ce qui suit quelques méthodes de description de la forme [BED, 11]:

A. Les moments orthogonaux

Les moments orthogonaux, comme leur nom l'indique, sont définis dans une base orthogonale, ce qui évite la redondance des informations portées par chacun des moments. Les deux types de moments orthogonaux les plus utilisés sont : les moments de Legendre et les moments de Zernike [BED, 11].

1 - Les **Descripteurs de Fourier DFs**: font partie des descripteurs les plus populaires pour les applications de reconnaissance de formes et de recherche d'images. Leur principe est de représenter le contour de l'objet par un signal 1D, puis de le décomposer en séries de Fourier.

a) Moments de Legendre

Les moments de Legendre sont définis à partir des polynômes du même nom. Ils sont définis dans le carré $[-1,1] * [-1,1]$ unité, ce qui oblige à normaliser l'objet dont on veut calculer ces moments .

✓ Le polynôme de Legendre d'ordre n est donné par :

$$\forall x \in [-1,1], \forall n \in N, P_n(x) = \frac{1}{2^n n!} * \frac{d^n (x^2 - 1)^n}{dx^n}$$

Les polynômes de Legendre $\{P_n(x)\}$ forment une base complète et orthogonale sur le domaine de définition $[-1,1]$.

$$\forall (x, y) \in [-1,1]^2, \forall (m, n) \in N^2, \iint_{-1}^1 P_m(x) P_n(y) dx dy = \frac{2}{2m+1} \delta_{mn}$$

❖ δ_{mn} Représente la fonction de Kronecker⁽¹⁾.

✓ Les moments de Legendre d'ordre N sont donc donnés par :

$$\forall (x, y) \in [-1,1]^2, \forall (p, q) \in N^2, N = p + q,$$

$$L_{pq} = \frac{(2p+1) * (2q+1)}{4} \iint_{-1}^1 P_p(x) P_q(y) f(x, y) dx dy$$

❖ Où $f(x, y)$ le niveau de gris d'un pixel de l'image I sur laquelle on calcule le moment.

① A partir de cette équation, on peut générer une infinité de moments de Legendre.

Plusieurs études sur la reconnaissance des formes ont démontré que l'utilisation des moments de Legendre de bas ordre (jusqu'à l'ordre 3) est suffisante pour représenter la forme globale de l'entité donnée [BED, 11].

b) Les moments de Zernike

Ce type des moments a été initialement introduit par Tague et qui sont construits à partir de polynômes complexes et forment un ensemble orthogonal complet définie sur le disque unité. Ils sont invariants par rotation et changements d'échelles et présentent des propriétés intéressantes sans termes de résistance aux bruits, efficacité informative et possibilité de reconstruction.

1 - **la fonction de Kronecker:** est une fonction de deux variables qui est égale à 1 si celles-ci sont égales, et 0 sinon.

- ✓ Les moments orthogonaux de Zernike d'ordre p sont définis de la manière suivante :

$$A_{mn} = \frac{m+1}{\pi} \iint I(x, y) [V_{m,n}(x, y)] dx dy$$

- ❖ Où m et n définissent l'ordre du moment et $I(x, y)$ le niveau de gris d'un pixel de l'image I sur laquelle on calcule le moment.

- ✓ Les polynômes de Zernike $V_{mn}(x, y)$ sont exprimés en coordonnées polaires :

$$V_{m,n}(r, \theta) = R_{m,n}(r) e^{-jn\theta}$$

- ❖ Où $R_{m,n}(r)$ est le polynôme radial orthogonal.

$$R_{m,n}(r) = \sum_{s=0}^{\frac{m-|n|}{2}} (-1)^s \frac{(m-s)!}{s! \left(\frac{m+|n|}{2} - s\right) \left(\frac{m+|n|}{2} - s\right)!} r^{m-2s}$$

- ❖ Avec $n = 0, 1, 2 \dots \infty$; $0 \leq |m| \leq n$ et $n - |m|$ un entier pair.

Les polynômes de Zernike sont orthogonaux, et donc les moments correspondants le sont également. Cette propriété d'orthogonalité annule l'effet de redondance de l'information portée par chaque moment) [ATE, 00].

B. les moments géométriques

Les moments géométriques permettent de décrire une forme à l'aide de propriétés statistiques. Ils représentent les propriétés spatiales de la distribution des pixels dans l'image [BED, 11]. La notion de moment en mathématiques (notamment en calcul des probabilités) a pour origine la notion de moment en physique.

- ✓ Le moment m_{pq} d'ordre $n = p + q$ d'une fonction de distribution $f(x, y)$ est définie par l'équation suivante :

$$m_{pq} = \iint_{-\infty}^{-\infty} x^p y^q f(x, y) dx dy$$

- ✓ Pour une image digitale de taille $M * N$, la formule ci-dessus devient :

$$m_{pq} = \sum_{y=0}^{N-1} \sum_{x=0}^{M-1} x^p y^q g(x, y)$$

- ❖ Où $g(x, y)$ l'intensité du point (x, y) .
- ✓ Les moments apportent différentes informations statistiques sur la forme :

↪ ordre 0, surface de la forme : m_{00} .

↪ ordre 1, centre de gravité de la forme, et est calculé par la formule [MME, 12]:

$$\bar{x} = \frac{m_{10}}{m_{00}} , \quad \bar{y} = \frac{m_{01}}{m_{00}}$$

Il est possible de calculer à partir de ces moments l'ellipse équivalente à l'objet. Afin de calculer les axes de l'ellipse, il faut ramener les moments d'ordre 2 au centre de gravité :

$$m_{20}^g = m_{20} - m_{00}\bar{x}^2$$

$$m_{11}^g = m_{11} - m_{00}\bar{x}\bar{y}$$

$$m_{02}^g = m_{02} - m_{00}\bar{y}^2$$

Puis on détermine l'angle d'inclinaison de l'ellipse α [BED, 11]:

$$\alpha = \frac{1}{2} \arctan \frac{2m_{11}^g}{m_{20}^g - m_{02}^g}$$

✓ Le moment centrale est défini par :

$$\mu_{pq} = \sum_{y=0}^{N-1} \sum_{x=0}^{M-1} (x - \bar{x})^p (y - \bar{y})^q g(x, y)$$

Les moments centrés sont utilisés pour le calcul des moments centrés normalisés : Ils sont calculés par l'expression suivante :

$$\eta_{pq} = \frac{\mu_{pq}}{s^{\frac{p+q}{2}+1}} ; p + q \geq 2$$

Où s représente la surface de l'objet [MME, 12].

C. Les invariants de Hu

Dans ce travail, l'accent sera mis sur les invariants de Hu, donc on va donner une vue plus détaillée sur ces descripteurs .

A partir des moments géométriques, Hu a proposé un ensemble de sept moments invariants aux translations, rotations et changement d'échelle. Ils sont très utilisés dans la littérature pour la description de formes en vue d'une classification ou d'une indexation.

a) L'historique de moment de Hu

L'histoire de moments invariants a commencé plusieurs années avant l'apparition des premiers ordinateurs, au 19e siècle dans le cadre de la théorie des invariants algébriques.

La théorie des invariants algébriques provient probablement du célèbre mathématicien allemand David Hilbert et a été soigneusement étudié aussi des Invariants Moment ont d'abord été présentés à la communauté de la reconnaissance des formes en 1962 par Hu, qui a utilisé les résultats de la théorie des invariants algébriques et tirait ses sept invariants célèbres de rotation des objets 2- Depuis amélioration et généralisation des invariants de Hu et aussi de son utilisation dans de nombreux domaines d'application [LAA, 14].

b) Les moments invariants de Hu

les moments invariants de Hu permettent de décrire la forme à l'aide de propriétés statistiques. Ils sont simples à manipuler, et sont robustes aux transformations géométriques comme la translation, la rotation et le changement d'échelle. Plusieurs techniques ont été développées pour la caractérisation et la représentation d'objets par ces moments. Hu a défini sept moments invariants issus de produits et quotients des moments centrés normés d'ordre 3, définis comme suit [MME, 12] [BED, 11].

Moments d'ordres 2:

$$\phi_1 = \eta_{20} + \eta_{02}$$

$$\phi_2 = (\eta_{20} + \eta_{02})^2 + 4\eta_{11}^2$$

Moments d'ordres 3:

$$\phi_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2$$

$$\phi_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2$$

$$\phi_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]$$

$$\phi_6 = (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03})$$

$$\phi_7 = (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + (3\eta_{12} - \eta_{30})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]$$

c) Caractéristiques des moments de Hu

Les six premiers moments sont invariants aux translations, aux changements d'échelles, et aux rotations ainsi qu'aux réflexions. Or, l'invariance aux réflexions peut être problématique quand il s'agit de reconnaître des images « miroirs ». C'est pourquoi M. K. Hu exploite aussi dans son système le 7^{ème} moment qui n'est pas invariant aux réflexions. Ce dernier change de signe lorsqu'une telle transformation est appliquée à l'image et permet donc de détecter celle-ci [MME, 12].

4.2. La description locale de l'image

Puisque les utilisateurs ont besoin de chercher des objets dans une image, le système doit utiliser les caractéristiques locales puisque les caractéristiques globales ne peuvent pas donner les résultats désirés toujours [VUN, 11].

On fait la segmentation de l'image pour la diviser en des régions locales. On souhaite que chaque région locale soit plus proche d'un objet dans l'image. Et puis on calcule les caractéristiques pour chacune des régions extraites. C'est difficile à choisir une "bonne" méthode de segmentation valide pour toutes les images.

4.3. Les descripteurs spécifiques (essentiellement biométrie)

Pour des applications où les images présentent moins de variance de l'une à l'autre, ces descripteurs ne sont pas suffisants, notamment dans des applications comme la reconnaissance de visages et la reconnaissance d'empreintes digitales. Des descripteurs spécifiques ont été proposés pour ces domaines particuliers en utilisant les connaissances expertes. Donc des descripteurs spécifiques ont été proposés pour ce domaine particulier en utilisant les connaissances expertes [TOU, 12].

I.3. Conclusion

Au long de ce chapitre, nous avons présenté les notions de base d'images numériques. Par la suite nous avons survolé sur ces différentes approches de description: descripteurs globale, locale et spécifique. Où nous avons vu les différentes méthodes et techniques connus dans la boîte à outils de la littérature pour le descripteur globale utilisé à ce travail.

Dans le prochain chapitre, nous allons présenter la carte auto-organisatrice (Carte de kohonen) qui je vais utiliser dans mon travail de classification des feuilles des plantes à l'aide des moments invariants de Hu.

Chapitre II

Les cartes auto-organisatrice (Carte de Kohonen)

Les cartes auto adaptatives ou auto organisatrices forment une classe de réseau de neurones artificiels fondée sur des méthodes d'apprentissage non-supervisées. Ce chapitre est consacré pour définir les différents concepts sur cette notion: l'idée de base, l'architecture et l'algorithme d'apprentissage.

II.1. Introduction

Dans ce travail je m'intéresse à la conception et la réalisation d'une application de classification automatique des images de feuilles des plantes. Cette application utilise la théorie des réseaux de neurones et plus précisément les cartes auto-organisatrices de Kohonen. Cette méthode de classification, construite à partir de l'algorithme de Kohonen dont l'apprentissage est non supervisé.

Dans ce chapitre je présente les cartes auto-organisatrices de Kohonen. Ensuite nous verrons l'architecture de la carte. En fin je présente l'algorithme d'apprentissage: principe et formalisation mathématique.

II.2. Réseaux de neurones

Un réseau de neurones artificiels est un modèle de calcul dont la conception est très schématiquement inspirée du fonctionnement des neurones biologiques.

Les réseaux de neurones sont généralement optimisés par des méthodes d'apprentissage de type probabiliste, en particulier bayésien. Ils sont placés d'une part dans la famille des applications statistiques, qu'ils enrichissent avec un ensemble de paradigmes permettant de créer des classifications rapides, et d'autre part dans la famille des méthodes de l'intelligence artificielle auxquelles ils fournissent un mécanisme perceptif indépendant des idées propres de l'implémenteur, et fournissant des informations d'entrée au raisonnement de logique formel [Web, 03].

L'une des caractéristiques les plus importantes des réseaux de neurones et leur capacité à apprendre, en effet, nous n'avons pas besoin d'une formulation rigoureuse d'un problème donné pour le résoudre, tout ce que nous avons besoin c'est d'une collection d'exemples représentative de la fonction désirée, ensuite le réseau s'adapte pour reproduire les sorties souhaitées quand un exemple d'entrée est présenté [BES, 00].

II.3. Cartes auto-organisatrices de Kohonen

Les cartes auto-organisatrices désignées par SOM (pour Self Organizing Maps) forment une classe de réseau de neurones artificiels fondée sur des méthodes d'apprentissage non-supervisées [Web, 04]. Les méthodes d'apprentissage non supervisé sont des méthodes consistant à inférer des connaissances sur des classes sur la seule base des échantillons d'apprentissage, et sans savoir a priori à quelles classes ils appartiennent [LAA, 14].

SOM ont été introduite par T. Kohonen en 1981 et sont inspirées comme beaucoup d'autres créations de l'intelligence artificielle, par la biologie. Il s'agit de reproduire le principe neuronal du cerveau des vertébrés : des stimuli de même nature excitent une région du cerveau bien particulière. Les neurones sont organisés dans le cortex de façon à interpréter tous les types de stimuli imaginables. De la même manière, la carte auto adaptative se déploie de façon à représenter un ensemble des données, et chaque neurone se spécialise pour représenter un groupe bien particulier des données selon les points communs qui les rassemblent [Web, 03].

Le principe des cartes de Kohonen est de projeter un ensemble complexe de données sur un espace de dimension réduite (2 ou 3). Cette projection permet d'extraire un ensemble de vecteurs dites référents ou prototypes. Ces prototypes sont caractérisés par des relations géométriques simples. La projection de données par SOM se produit tout en conservant la topologie et les métriques les plus importantes des données d'entrées lors de l'affichage, c'est-à-dire les données proches (dans l'espace d'entrée) vont avoir des représentations proches dans l'espace de sortie et vont donc être classés dans le même cluster ou dans des clusters voisins [SOU, 14].

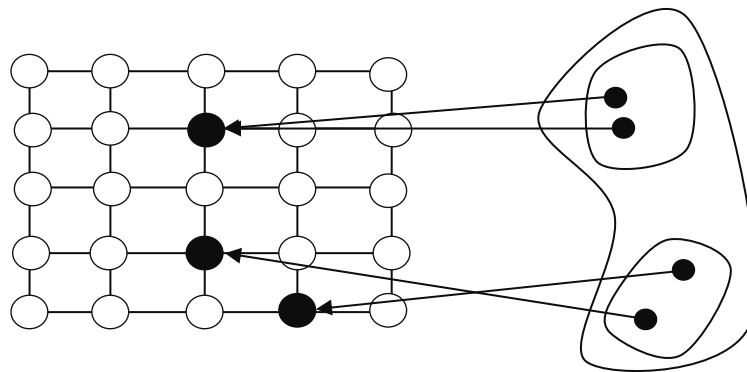


Fig II.1: Des neurones voisins sur la carte représentent des objets assez "proche" dans l'espace des données d'entrées

II.3.1. Architecture des cartes de Kohonen

Le SOM est une grille (ou réseau) de dimension faible qui contient un nombre M de neurones (ou nœuds, ou unités). Il est habituellement composé de deux couches de neurones, une couche d'entrée et une couche de sortie. Dans la couche d'entrée, chaque objet à classer (dans notre cas, les moments hu des images) est représenté par un vecteur multidimensionnel. La couche (topologique) d'adaptation ou la couche de sortie est composée d'un treillis de neurones selon la géométrie prédéfinie. Chaque neurone de la couche topologique est totalement connecté aux vecteurs de la couche d'entrée. Les vecteurs poids de ces connexions

forment le vecteur référent ou prototype associé à chaque neurone, qui est de même dimension que les vecteurs d'entrées [SOU, 14].

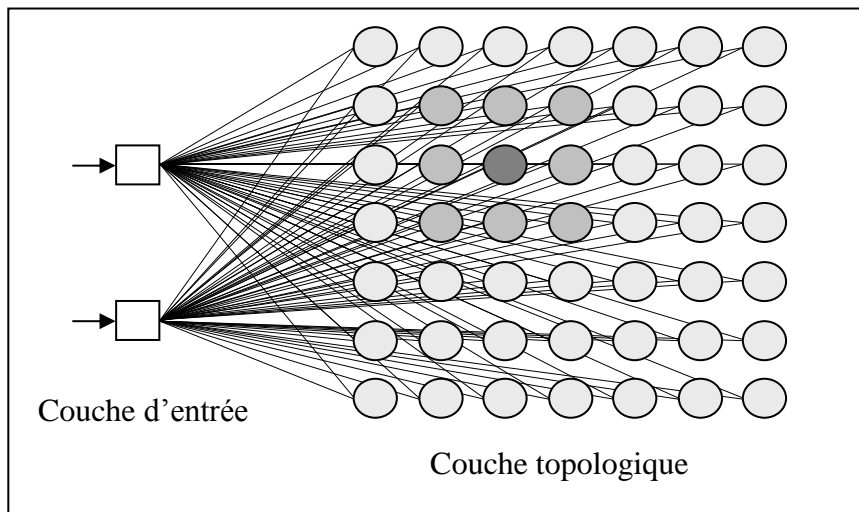


Fig II.2: Structure d'une carte auto-organisatrice.

La structuration topologique de la couche provient d'une mesure de la distance définie sur les neurones du réseau. Ainsi, on définira le voisinage immédiat $V_{c,1}$ d'un neurone $i \in S$ comme l'ensemble des neurones au plus a une distance 1 de $c \in S$:

$$V_{c,1} = \{d(i, c) \leq 1, i \in S\}$$

❖ Où S est grille neuronal.

A partir de cette définition élémentaire, on nommera voisinage Λ étendu l'ensemble $V_{c,m}$ des neurones à une distance inférieure ou égale à m de neurone $c \in S$:

$$V_{c,m} = \{d(i, c) \leq m, i \in S\}$$

Plusieurs formes de cartes sont possibles, selon le nombre et la disposition du neurone qui constitue ce voisinage. En général, les cartes s'articulent 1,2 ou 3D ; la carte la plus employée dispose ses neurones en voisinage rectangulaire ou hexagonal en 2D (voir fig I.3).

Cette structuration n'est pas explicitement présente dans l'architecture du réseau lors de sa conception. En effet, tout l'intérêt de ce type de réseau est que la notion de voisinage se développe progressivement par apprentissage, et n'est apparent que dans le comportement des neurones du réseau [HAM, 03].

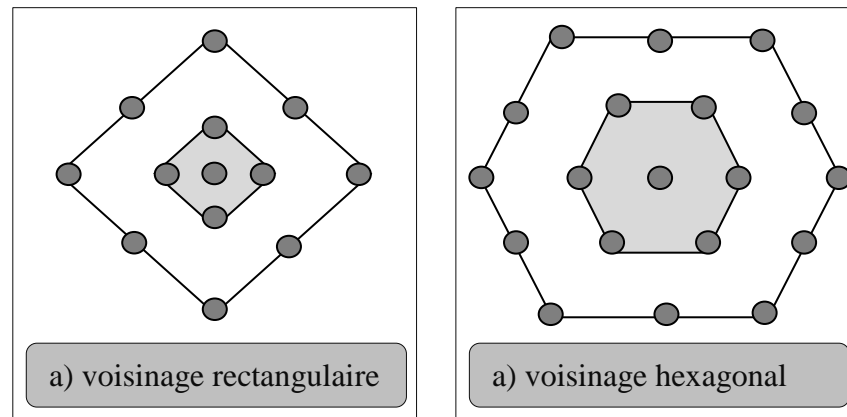


Fig II.3: Les voisinages les plus utilisés des cartes de kohonen: (a) rectangulaire (b) hexagonal.

II.3.2. La règle d'apprentissage

Le mode d'apprentissage des réseaux auto-organisés constitue le principal volet d'innovation de ceux-ci puisque leur architecture représente plus ou moins des combinaisons d'autres types de réseau comme les mémoires associatives, le perceptron, le perceptron multicouche. Il consiste principalement à trouver le neurone gagnant i à l'aide d'une mesure de similarité et de mettre à jour les poids des neurones concernés c'est-à-dire ceux qui se trouvent à l'intérieur du voisinage topologique $\Lambda_i(n)$.

Une étape de l'apprentissage s'effectue en choisissant aléatoirement un vecteur de l'ensemble d'apprentissage x . Ce vecteur est ensuite comparé à chacun des vecteurs poids. Le neurone gagnant $i(x)$ est ensuite identifié à l'aide d'une mesure de similarité habituellement définie connue étant la distance euclidienne entre les deux vecteurs.

$$i(x) = \arg \min \|x - \omega_j\| \quad j = 1, 2, \dots, N$$

❖ Où N est le nombre de neurones de grille.

Et $\| \cdot \|$ est la norme euclidienne telle que définie par:

$$\|X\| = \sqrt{\sum_{i=0}^{i=n} (x_i)^2}$$

Une fois trouvé le vecteur de poids gagnant, le réseau s'ajuste de manière à mieux représenter ce vecteur d'apprentissage. Le résultat final peut être visualisé par un filet s'ajustant au nuage des données d'entrée de façon à ce que chaque vecteur poids approxime la densité de probabilité d'une région de l'ensemble d'apprentissage.

La règle d'apprentissage pour un neurone j appartenant au voisinage topologique variable du neurone gagnant $i(x)$ décrit une rotation des vecteurs poids vers le vecteurs d'entrée et s'énonce comme suit:

$$\omega_j(n+1) = \begin{cases} \omega_j(n) + \eta(n)[x - \omega_j(n)] & \text{ssi } j \in \Lambda_{i(x)}(n). \\ \omega_j(n) & \text{ssi } j \notin \Lambda_{i(x)}(n). \end{cases}$$

Où $\Lambda_{i(x)}(n)$ dénote le voisinage topologique autour du neurone gagnant au temps discret n . et $\eta(n)$ est la fonction de voisinage qui contient une fonction de taux d'apprentissage $\alpha(n)$. elle détermine aussi le taux d'adaptation des neurones situés à l'intérieur de $\Lambda_{i(x)}(n)$. Cette fonction se veut un moyen plus simple de simuler les connexions latérales entre neurones elle s'exprime comme suit.

$$\alpha(n) = \alpha_0 \cdot \exp\left(-\frac{n}{N}\right)$$

❖ Où N est le nombre totale d'itérations.

Dans le cas d'une fonction de voisinage de Gaussienne autour du neurone gagnant $i(x)$, $\eta(n)$ s'exprime comme suit:

$$\eta(n) = \alpha(n) \cdot \exp\left(-\frac{\|r_j - r_{i(x)}\|^2}{2\sigma(n)^2}\right)$$

Où $\|r_j - r_{i(x)}\|$ est la distance entre le neurone j et celui gagnant $i(x)$ [HUG, 98]. Et σ s'appelle coefficient de voisinage. Son rôle est de déterminer un rayon de voisinage autour du neurone gagnant s'exprime comme suit:

$$\sigma(n) = \sigma_0 \cdot \exp\left(-\frac{n}{\lambda}\right)$$

Où σ_0 dénote la largeur du grille à l'instant $n=0$, et la valeur de λ dépend de σ_0 et le nombre d'itérations n de l'algorithme choisi pour exécuter s'exprime comme suit [Web, 05]:

$$\lambda = n/\log(\sigma_0)$$

II.3.3. Algorithme d'apprentissage d'une carte de Kohonen

L'algorithme de Kohonen a tout d'abord été introduit comme un algorithme d'apprentissage visant à modéliser les mécanismes de mise en place d'associations "respectant les voisinages" qu'on peut mettre en évidence dans le système nerveux biologique.

Mais en fait, dans le contexte de l'analyse des données, cet algorithme apparaît comme une méthode de classification particulièrement intéressante, extension de la version

stochastique de la méthode des centres mobiles. Sa propriété essentielle est qu'il permet de définir une notion de voisinage entre les classes, et qu'il fournit par conséquent des représentations graphiques et des visualisations très parlantes de données tant quantitatives que qualitatives [HAM, 03].

L'algorithme d'apprentissage se déroule en plusieurs étapes et sur plusieurs itérations comme suit:

Algorithme de traitement

1- Initialisation

la grille doit être initialisée de manière aléatoire.
donner une petite valeur positive au taux d'apprentissage α .

2- Présentation

tirage aléatoire d'une entrée x dans la base d'apprentissage afin de le présenter au réseau.

3- Similarité

recherche du neurone gagnant $i(x)$ en utilisant le critère de la distance euclidienne minimale (neurone plus proche)

4- Apprentissage

Mettre à jour les poids des neurones appartenant au voisinage topologique $\Lambda_{i(x)}(n)$ suivant le pas d'adaptation $\eta(n)$.
Adapter les paramètres d'apprentissage $\Lambda_{i(x)}(n)$ et $\eta(n)$ pour un meilleur résultat.

5- Itération

Retourner à l'étape 2 jusqu'à ce que le critère de distance euclidienne minimale soit atteint ou jusqu'à ce que les poids convergent.

II.3.4. Domaines d'application

Les cartes de Kohonen permettent le classement d'objets sur lesquels aucune information relative à leur classification n'est disponible a priori. Une fois le réseau organisé (une fois l'apprentissage effectué), chaque objet est classé dans une zone qui correspond à son « gagnant ».

L'utilisation des cartes de Kohonen dans la recherche documentaire consiste à présenter l'ensemble des documents à la carte pour les classer. Ensuite représenter le classement sous forme graphique compréhensible à l'être humain et qui permettra à l'utilisateur de naviguer dans le fond documentaire pour rechercher l'information désirée [HAM, 03].

II.3.5. Les avantages de cartes de Kohonen

Les SOM représentent plusieurs avantages, parmi lesquels on citera :

- ↪ La possibilité d'une visualisation graphique des résultats.
- ↪ La représentation des données en plusieurs dimensions.
- ↪ La facilité à comprendre ce modèle.
- ↪ Leur capacité à classifier les corpus d'une façon non supervisée, et cela, en utilisant la fonction de densité de l'échantillon [HIL, 09].

II.3.6. Inconvénients des cartes de Kohonen

Malgré les grands avantages que représentent les SOM, ils ont aussi des limites qu'on peut résumer dans les points suivants :

- ↪ Leur temps énorme de convergence.
- ↪ La représentation des données dans ce type de méthodes n'est pas unique.
- ↪ Il n'est pas sûr que ce type d'algorithmes converge dans un espace multidimensionnel [HIL, 09].

II.4. Conclusion

Les cartes d'auto-organisation constituent une alternative intéressante aux modèles classiques pour résoudre les problèmes de représentation et de synthèse. Au long de ce chapitre, nous avons présenté les différentes technologies derrière cette notion tel que: l'architecture, la règle d'apprentissage et l'algorithme de Kohonen.

J'ai adopté les moments invariants de Hu comme descripteur utilisé pour mon travail de classification des images de feuilles des plantes à l'aide de cartes d'auto-organisation. L'étape suivante l'efficacité de notre choix.

Chapitre III

La conception

Dans ce chapitre, je fais la conception et la modélisation de l'application en utilisant le langage UML .Cependant mon application n'est pas d'une grande complexité afin d'utiliser la totalité des diagrammes d'UML, j'utiliserai que les diagrammes que je trouve nécessaires.

III.1. Introduction

Dans ce chapitre je présenter une conception pour assurer un réalisation succès de mon application en utilisant le langage UML. Cependant mon application n'est pas d'une grande complexité pour d'utiliser la totalité des diagrammes d'UML, nous n'utiliserons que les diagrammes que nous trouverons nécessaires.

Dans ce cas, je commence par le diagramme de cas utilisation, puis pour chaque cas d'utilisation un diagramme de séquences, enfin un diagramme de classe.

III.2. Présentation de langage UML

UML (Unified Modeling Language, que l'on peut traduire par "langage de modélisation unifié") est une notation permettant de modéliser un problème de façon standard. Ce langage est né de la fusion de plusieurs méthodes existant auparavant, est devenu désormais la référence en terme de modélisation objet, à un tel point que sa connaissance est souvent nécessaire pour obtenir un poste de développeur objet [GRA, 98].

UML permet à des programmeurs utilisant des langages de programmation différents de parler la même langue. De plus, UML est un langage simple et intuitif. Elle est constitué de 13 diagrammes qui utilisent des symboles sémantique bien précise. Ils permettent de concevoir une modélisation fiable limitant ainsi beaucoup les grosses erreurs de programmation (erreurs de conception).

III.3. La conception de l'application

Le principe général de mon travail de classification des feuilles des plantes se déroule en deux phases. La première phase est une phase d'extraction des caractéristiques, se base sur le descripteur globale afin d'utiliser les moments invariants de Hu. La deuxième utilise les moments invariants de Hu comme entrées à la méthode carte auto-organisatrice de KOHONEN pour effectuer l'apprentissage et faire la classification.

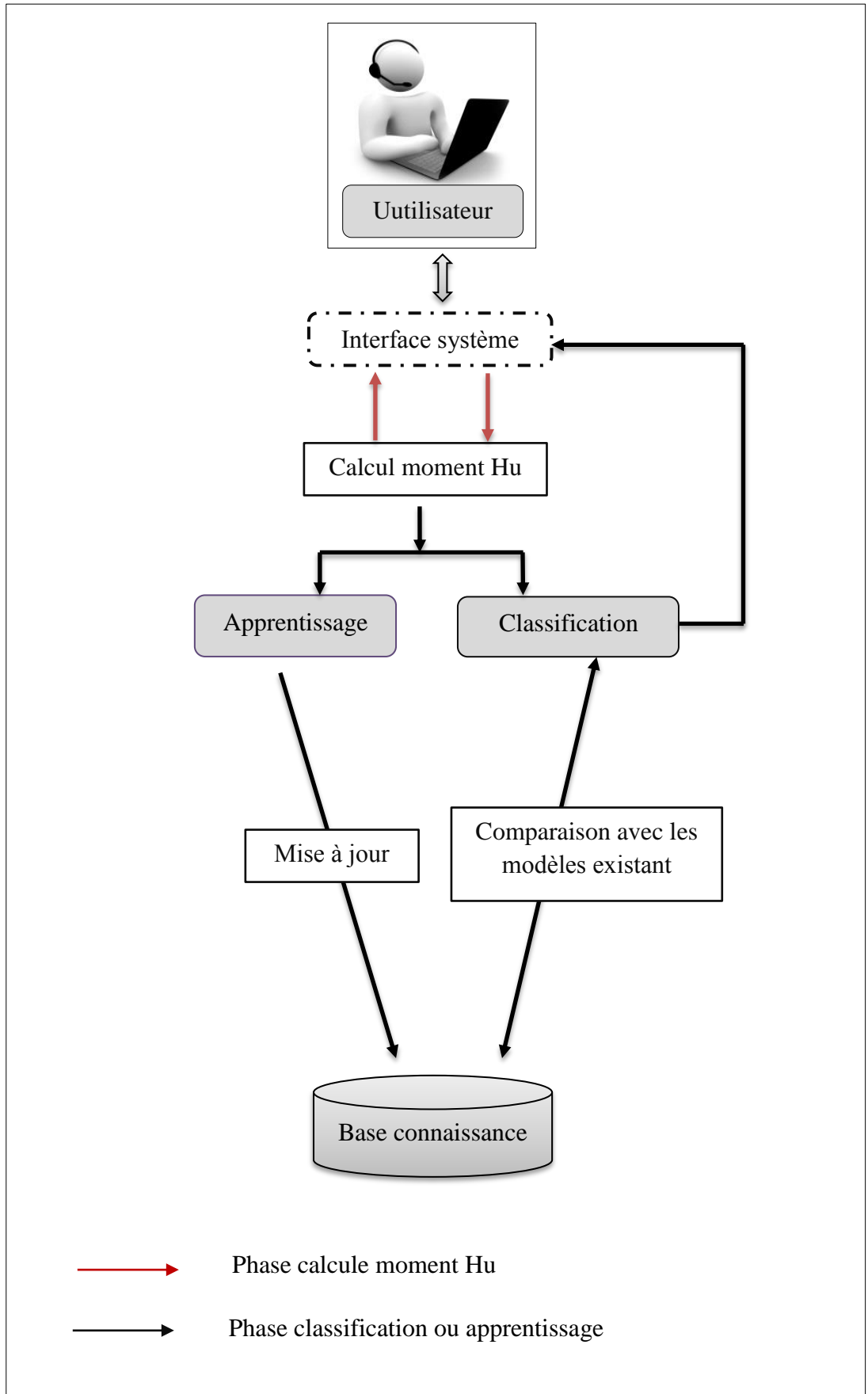


Fig III.1: Architecture conceptuelle du système.

III.4. Les diagrammes utilisés

III.4.1. Le diagramme des cas d'utilisation

L'ensemble des cas d'utilisation décrit exhaustivement les exigences fonctionnelles et techniques du système. Chaque cas d'utilisation correspond donc à une fonction métier du système, selon le point de vue d'un de ses acteurs.

Dans notre application l'utilisateur fournit une image dans le but de lui répondre par leur type ou classe. L'application permet aussi de calculer le moment Hu, et d'ajoute un nouveau classe de plantes.

Les cas d'utilisation de mon système sont représentés par la figure III.1 suivante:

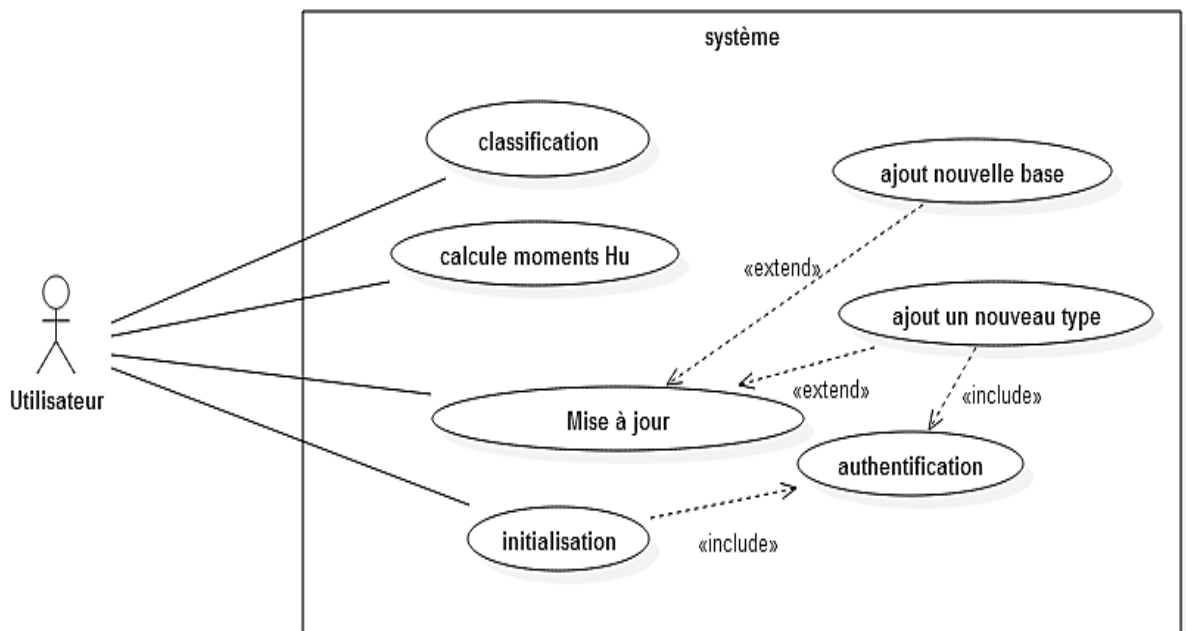


Fig III.2: Diagramme Cas d'utilisation.

III.4.2. Diagramme de séquences

les diagrammes de séquences permettent de représenter des collaborations entre objets selon un point de vue temporel et peuvent servir à illustrer des cas d'utilisation. Donc, on y met l'accent sur la chronologie des envois de messages.

1.2.4. Le cas d'utilisation «classification»



Fig III.3: cas d'utilisation «classification»

✓ Diagramme de séquences

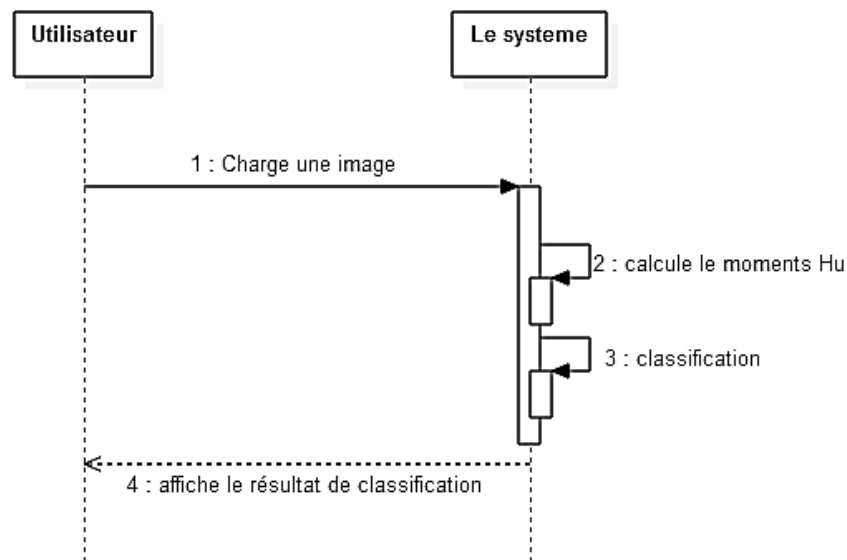


Fig III.4: diagramme de séquences «classification»

✓ Le scénario

- L'utilisateur charge une image.
- Le système calcule les moments Hu de cette image.
- Le système appelle la méthode de classification carte auto-organisatrice de KOHONEN et fait la classification.
- Le système affiche le résultat de classification .

2.2.4. Le cas d'utilisation «calcul moment Hu»



Fig III.5: cas d'utilisation «calcul moment Hu»

✓ Diagramme de séquences

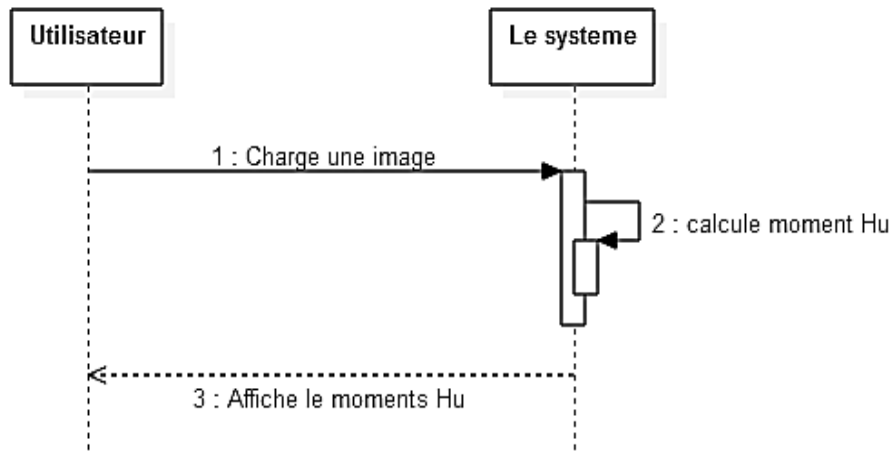


Fig III.6: diagramme de séquences «calcul moment Hu»

✓ Le scénario

- L'utilisateur charge une image.
- Le système calcule les moments Hu de cette image.
- Le système affiche les moments Hu de cette image.

2.2.5. Le cas d'utilisation «Mise à jour»

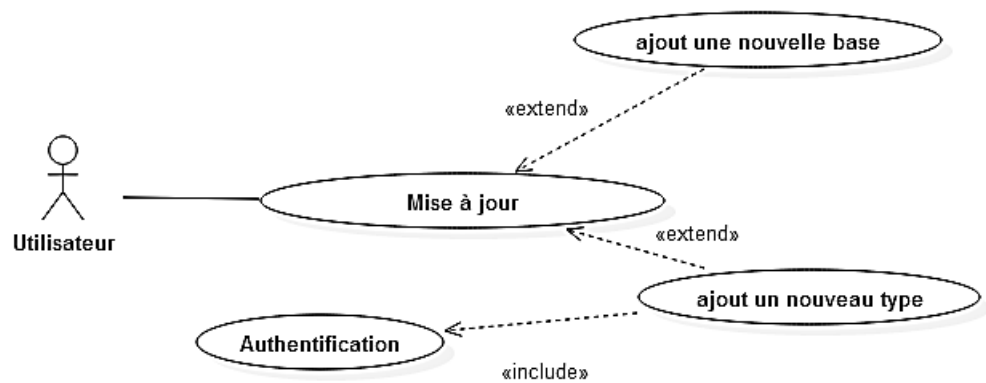


Fig III.7: cas d'utilisation «Mise à jour».

✓ Diagramme de séquences «ajout un nouveau type»

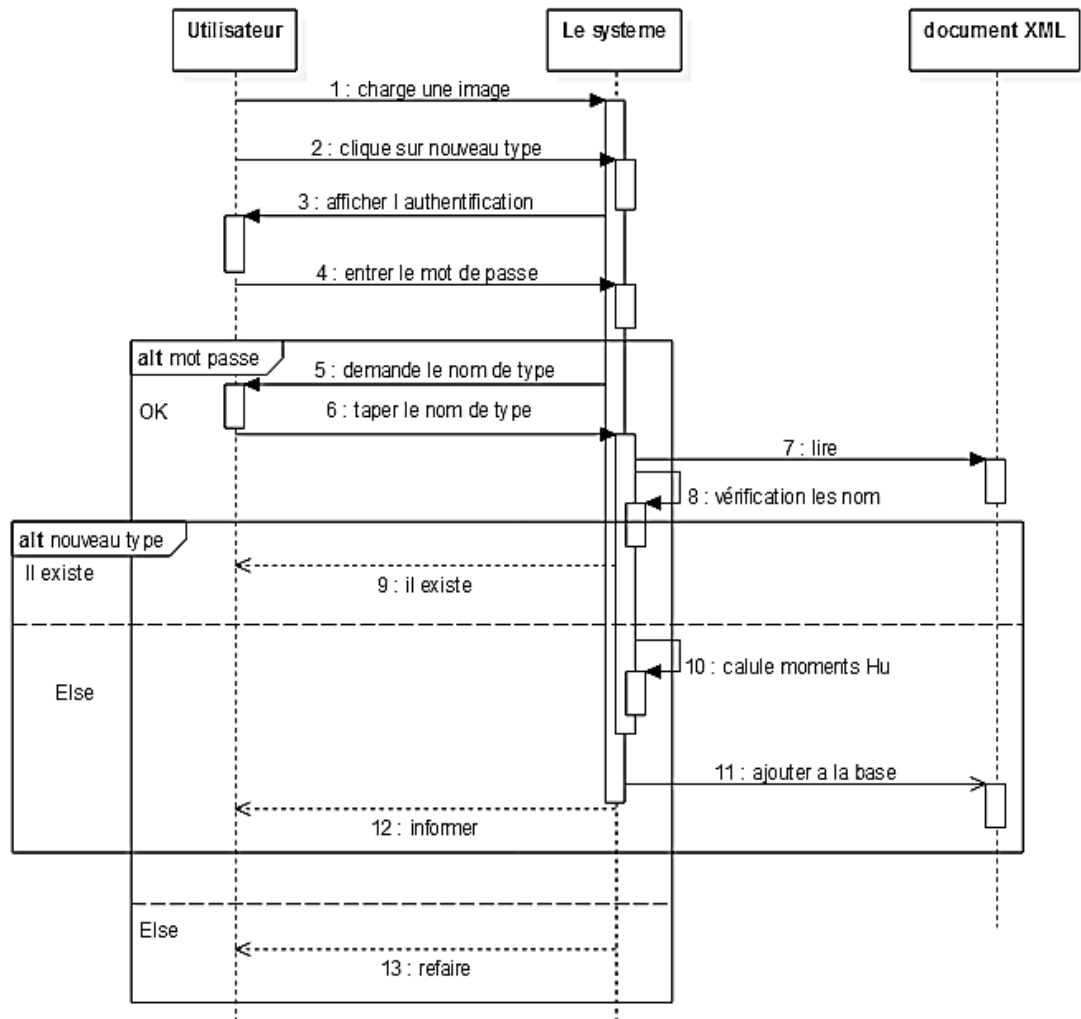


Fig III.8: diagramme de séquences «ajout un nouveau type»

✓ Le scénario

- L'utilisateur charge une image et clique sur nouveau type .
- Le système affiche l'authentification .
- L'utilisateur entre mot de passe .
- Si le mot de passe invalide le système demande refaire
- Sinon le système demande le nom de nouveau type.
- L'utilisateur tape le nom de nouveau type.
- Le système vérifie le type existant .
- Si cet nouveau type existe le système informe.
- Sinon le système calcule moment Hu et ajoute le nouveau type a la base d'image et informe.

✓ Diagramme de séquences «ajout une nouvelle base»

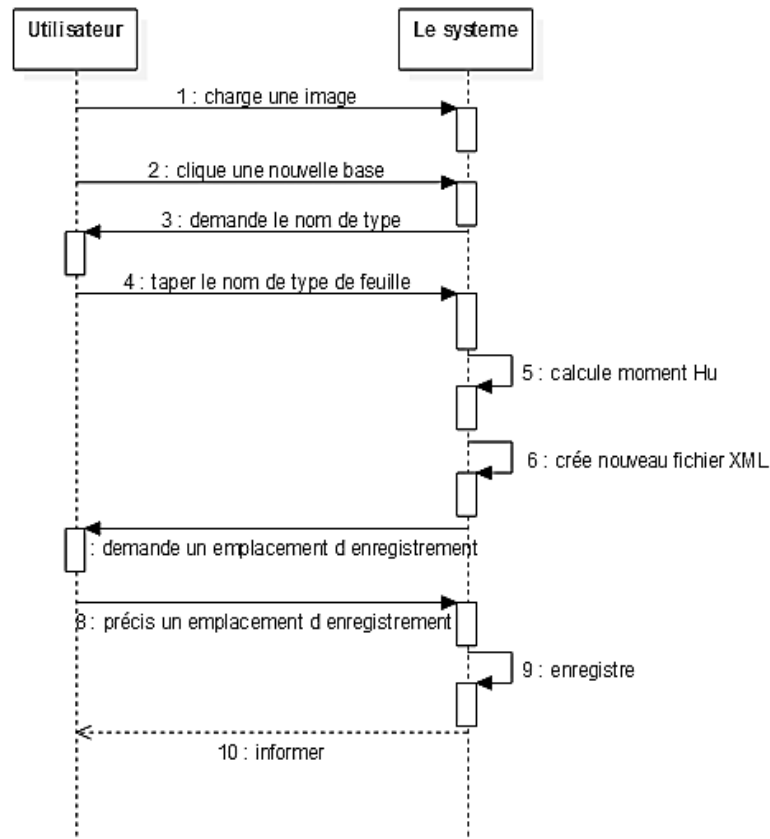


Fig III.9: diagramme de séquences «ajout un nouvelle base».

✓ Le scénario

- L'utilisateur charge une image.
- L'utilisateur clique une nouvelle base .
- Le système demande le nom de type .
- L'utilisateur tape le nom de type de feuille.
- Le système calcule les moments Hu de cette image.
- Le système crée un nouvelle base des image sous la forme fichier XML
- Le système demande l'emplacement local pour sauvegarde le fichier XML
- L'utilisateur précise l'emplacement pour l'enregistrement .
- Le système enregistre le fichier et informe.

2.2.6. Le cas d'utilisation «Initialisation»



Fig III.10: cas d'utilisation «Initialisation».

✓ Diagramme de séquences «initialisation»

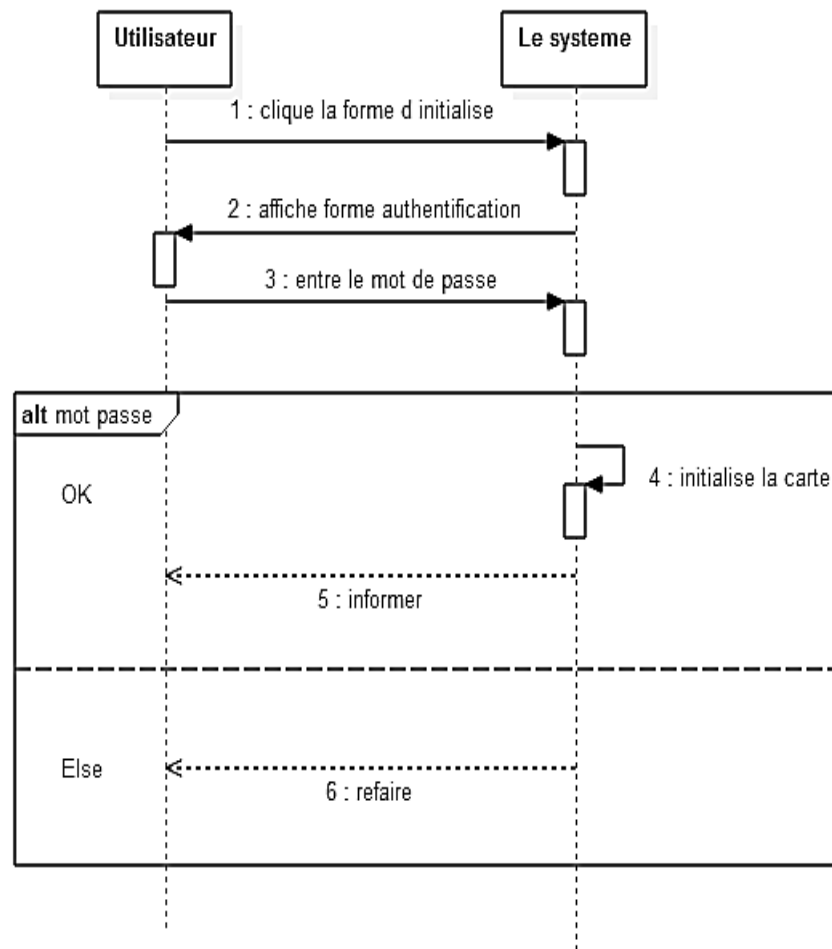


Fig III.11: diagramme de séquences «initialisation»

✓ Le scénario

- L'utilisateur clique sur initialisation.
- Le système demande le mot de passe.
- Si le mot de passe valide le système initialise la carte Kohonen et informé.
- Sinon le système calcule demande refaire.

III.4.3. Le diagramme de classes

Un diagramme de classes fournit une vue globale statique d'un système en présentant ses classes, interfaces, et les relations entre elles. Donc le figure de suivant, présente la classe principal du notre . système:

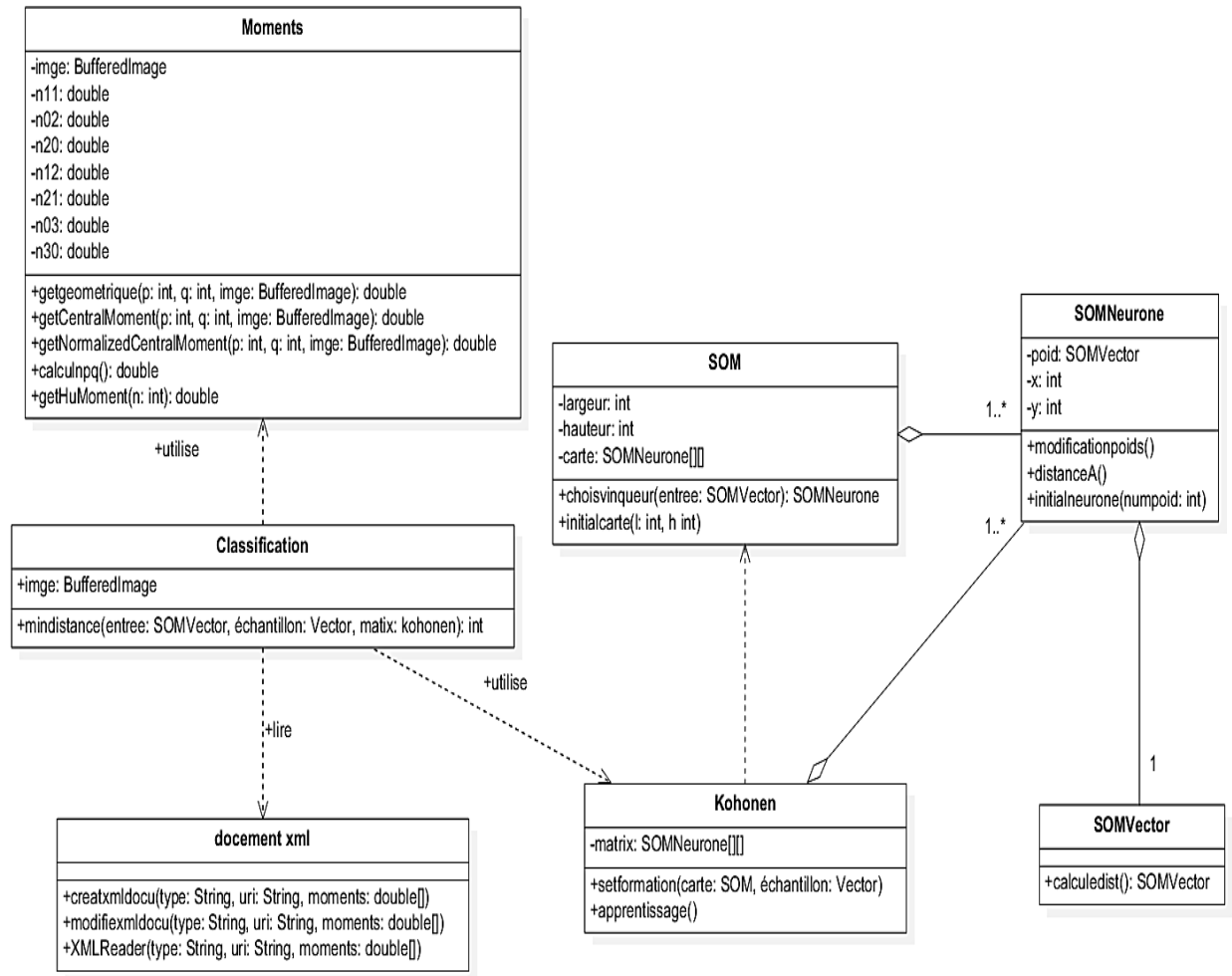


Fig III.12: Diagramme de classe.

III.5. Conclusion

Dans le but de faciliter l'implémentation de notre système de classification des feuilles des plantes, j'ai présenté dans ce chapitre une capture des diagrammes utilisés lors de la modélisation.

Les résultats de ce chapitre seront enrichis par des détails d'implémentation dans le chapitre suivant pour réaliser notre système de recherche.

Chapitre IV

Réalisation

Dans ce chapitre, consacré à la réalisation et la mise en œuvre de l'application de classification Des feuilles de plantes, je vais présenter les outils de développement adoptés et les principales interfaces et fenêtres de l'application.

IV.1. Introduction

Après avoir présenté dans le chapitre précédent la conception. Je vais présenter dans ce qui suit la mise en œuvre de mon application. je commence tout d'abord par une présentation d'outil de programmation choisi tel que le langage de programmation, ainsi que les motivations de ce choix. Ensuite nous allons décrire l'interface de notre application.

IV.2. les outils de développement

IV.2.1. Environnement de travail

Pour réaliser notre application, nous avons utilisé l'environnement Netbeans 7.4. il est un environnement de développement intégré ,placé en open source par Sun en juin 2000 sous licence CDDL et GPLv2 (Common Development and Distribution License). En plus de Java, NetBeans permet également de supporter différents autres langages, comme C, C++, JavaScript, XML, Groovy, PHP et HTML de façon native ainsi que bien d'autres (comme Python ou Ruby) par l'ajout de greffons. Il comprend toutes les caractéristiques d'un IDE moderne (éditeur en couleur, projets multi-langage, refactoring, éditeur graphique d'interfaces et de pages Web).

NetBeans constitue par ailleurs une plate-forme qui permet le développement d'applications spécifiques (bibliothèque Swing (Java)). L'IDE NetBeans s'appuie sur cette plate-forme [Web, 06].

IV.2.2. Le langage de programmation

Le langage Java est un langage de programmation informatique orienté objet créé par James Gosling et Patrick Naughton, employé de Sun Microsystems, avec le soutien de Bill Joy (cofondateur de Sun Microsystems en 1982), présenté officiellement le 23 mai 1995 au SunWorld.

La particularité et l'objectif central de Java est que les logiciels écrits dans ce langage doivent être très facilement portables sur plusieurs systèmes d'exploitation tels que UNIX, Windows, Mac OS ou GNU/Linux, avec peu ou pas de modifications. Pour cela, divers plateformes et frameworks associés visent à guider, sinon garantir, cette portabilité des applications développées en Java [Web, 07].

IV.2.3. Présentation XML

L'Extensible Markup Language (XML, « langage à balise extensible » en français) est un langage informatique de balisage générique qui dérive du SGML. Cette syntaxe est dite « extensible » car elle permet de définir différents espaces de noms, c'est-à-dire des langages avec chacun leur vocabulaire et leur grammaire comme XHTML, XSLT, RSS, SVG.... Elle est reconnaissable par son usage des chevrons (< >) encadrant les balises. L'objectif initial est de faciliter l'échange automatisé de contenus complexes (arbres, texte riche...) entre systèmes d'informations hétérogènes. Avec ses outils et langages associés, une application XML respecte généralement certains principes :

- ↪ la structure d'un document XML est définie et validé par un schéma ;
- ↪ un document XML est entièrement transformable dans un autre document XML [Web, 08].

IV.3. Description de l'application

J'ai essayé de créer une interface graphique qui cache le plus possible des détails d'exécution de mon application à fin d'offrir une utilisation simple et conviviale pour l'utilisateur de cet outil.

Après l'exécution de l'application, la fenêtre principale "Welcome" va apparier. Cette fenêtre reçoit les utilisateurs:



Fig IV.1: L'interface de notre application.

Le bouton "ENTRER" de cette interface permet de lancer les autres composants et d'afficher une nouvelle fenêtre "Classification" qui permet de parcourir toutes les fenêtres comme suit:

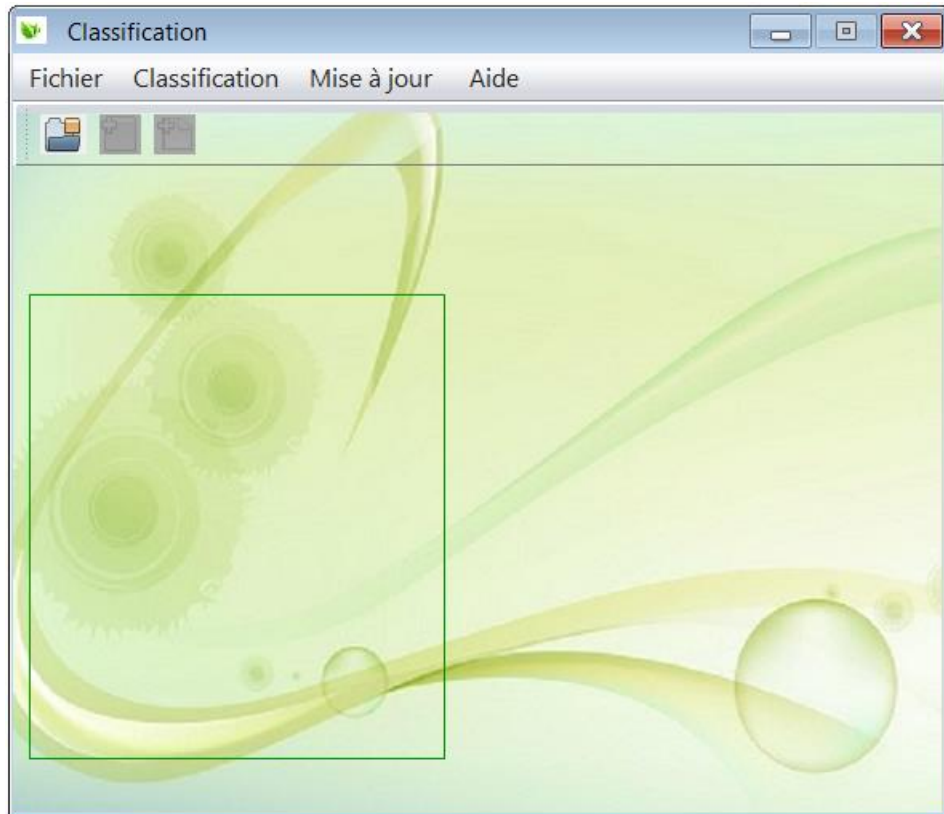


Fig IV.2: La fenêtre classification.

Cette fenêtre contient un Menu qui permet de parcourir fichiers, Classification, Mise à jour et autres comme initialisation, About ... etc.

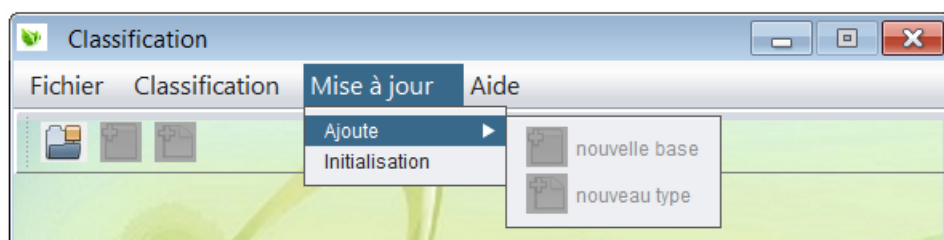


Fig IV.3: interface avec le Menu.

- ✓ **Fichier** : Ce item offre la possibilité d'effectuer les opérations suivantes :
 - ↗ Ouvrir: pour charger une image.
 - ↗ Quitter: pour quitter l'application.

- ✓ **Ouvrir:** Le clic sur item "ouvrir" permet d'afficher une nouvelle boîte de dialogue pour charger l'image.

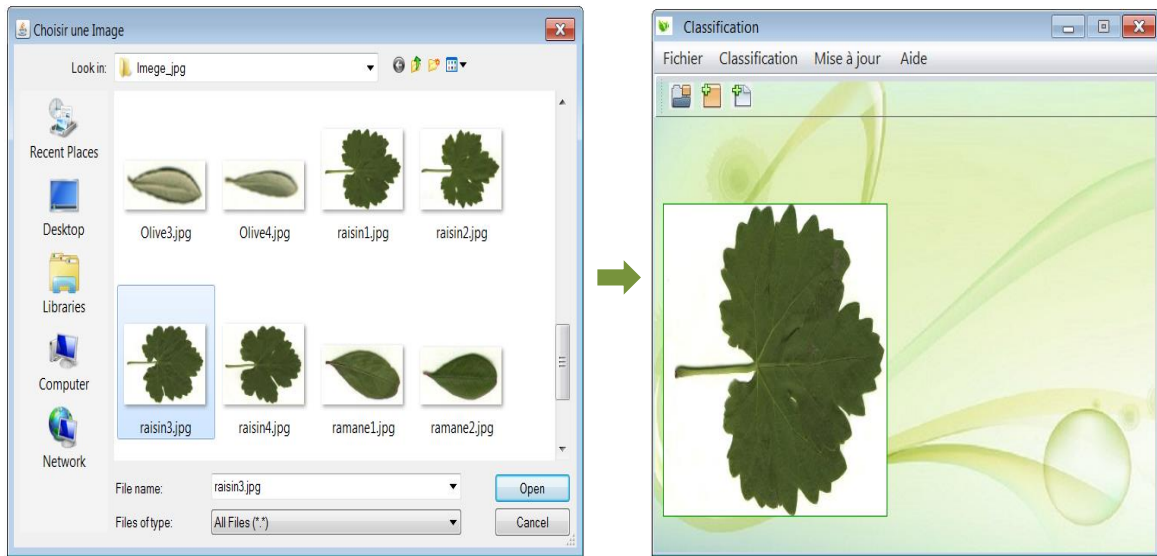


Fig IV.4: Charge un image.

- ✓ **Classification :** Ce item offre la possibilité d'effectuer les opérations suivantes :
 - ↪ Calcule moment Hu.
 - ↪ Classification.
- ✓ **Classification:** le clic sur ce item ouvre la fenêtre Classification.



Fig IV.5: Interface de résultats de Classification.

- ✓ **Calcul moments de Hu:** clic sur ce item lancé le calcule automatiquement les moments de Hu et affiche le résultat:

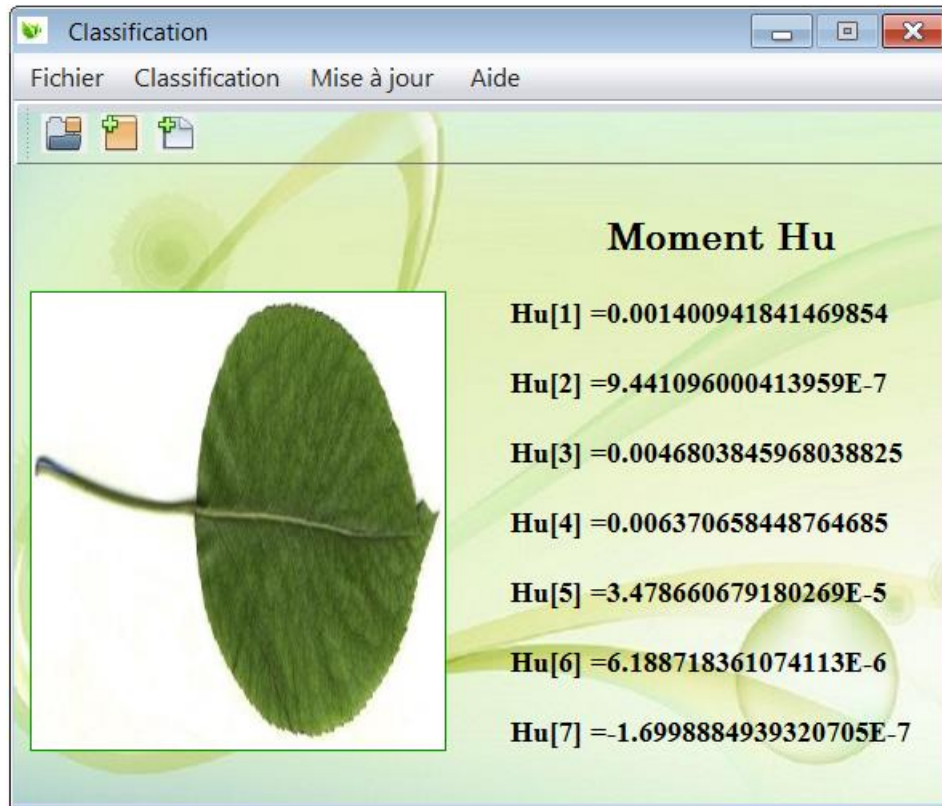


Fig IV.6: la résultat de calcule moment Hu.

- ✓ **Mise à jour:** Ce item offre la possibilité d'effectuer les opérations suivantes.
 - ↪ Initialisation;
 - ↪ Ajouté un nouveau type;
 - ↪ Ajouté une nouvelle base d'image.
- ✓ **Initialisation:** Ce item effectue l'initialisation de carte Kohonen. Quand on clique à ce item l'application demande l'authentification. Si le mot passe juste il va initialise la carte.

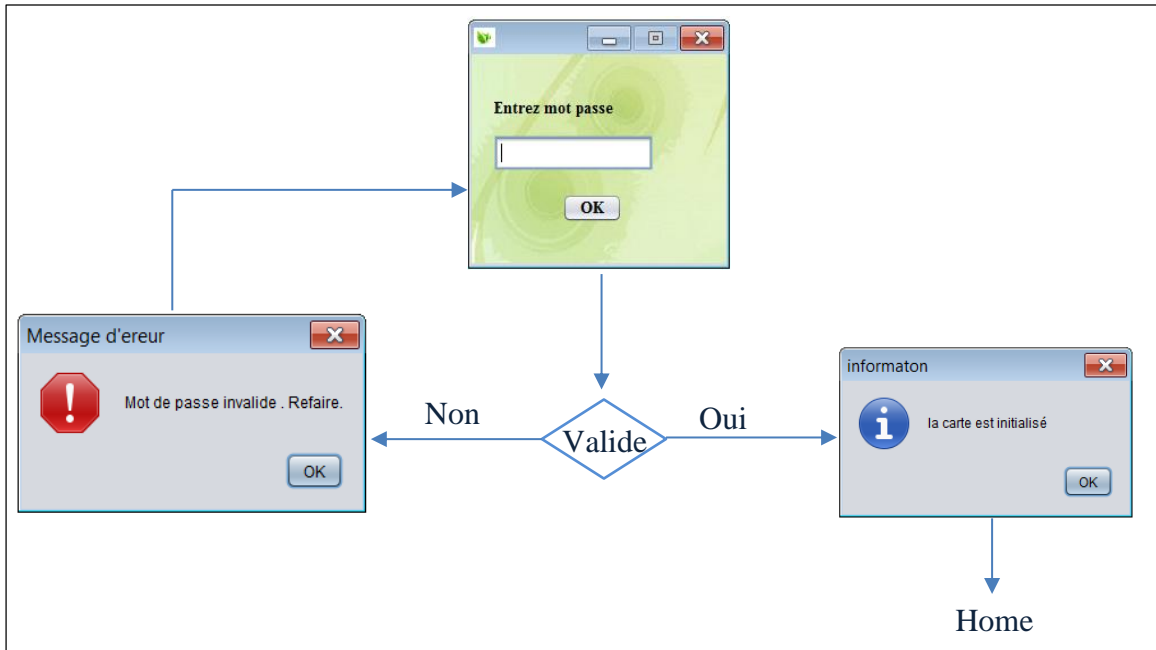


Fig IV.7: les interfaces initialisation

- ✓ **Ajouté un nouveau type:** Ce item effectue et ajoute un nouveau type. Quand on clique à ce item le système demande l'authentification. Si le mot passe juste il va permet d'ajoute nouveau type.

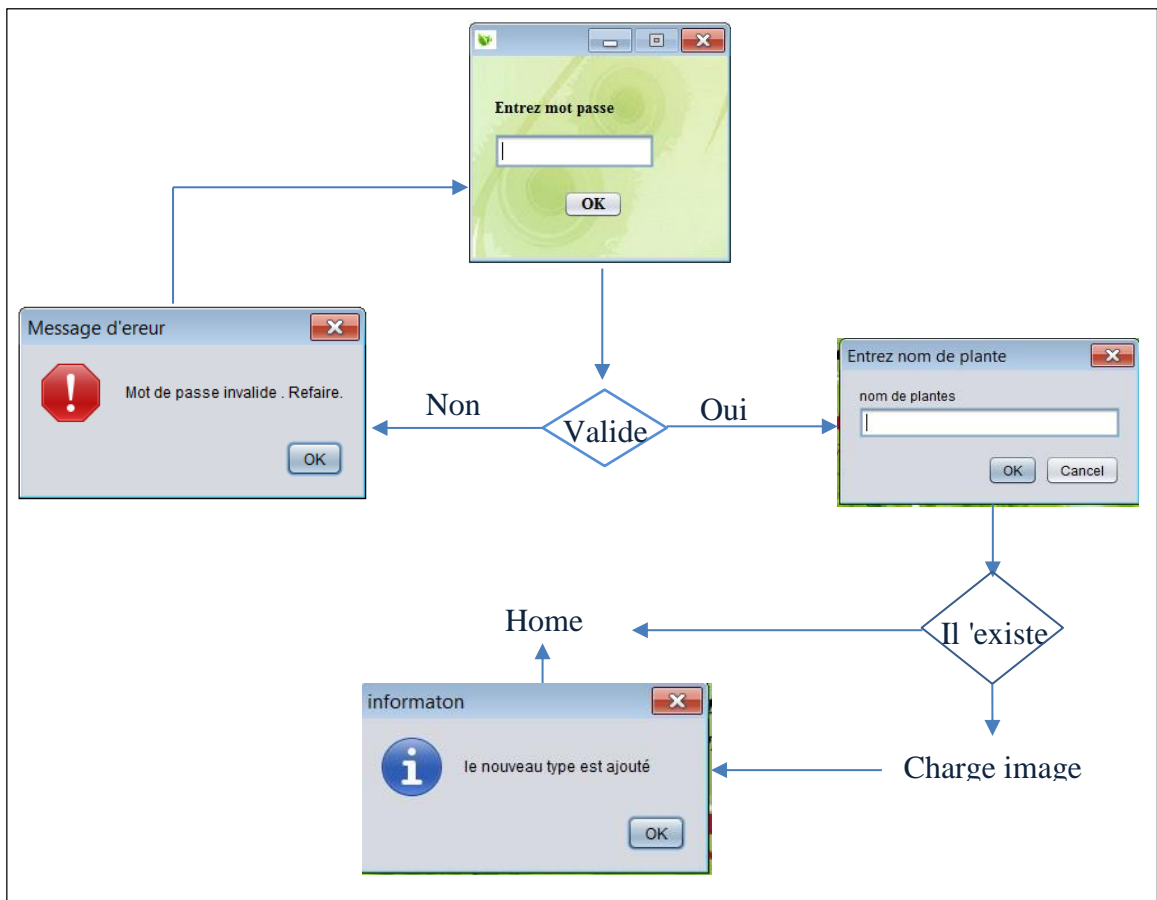


Fig IV.8: les interfaces ajout nouveau type.

✓ **Aide:** contient les deux menus items suivants :

↳ Aide: Affichage de l'aide de l'application

↳ A propos: Affichage des information sur l'étudiant qui développe cette application.

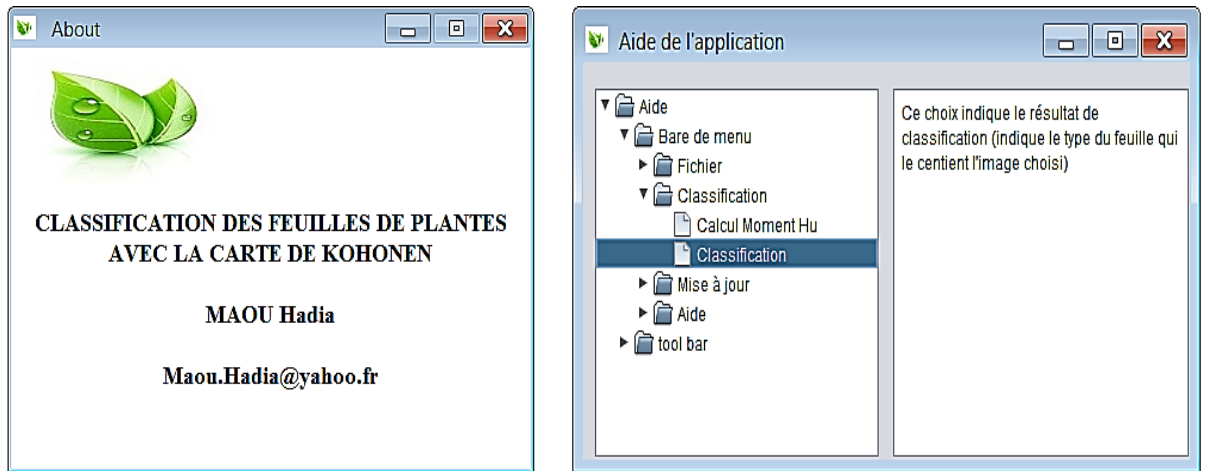


Fig IV. 9: l'interface d'item Aide.

IV.4. Test et Résultats

Pour pouvoir jugé la performance du système j'ai utilisé un ensemble d'exemples de feuille. Ces ensemble sont divisé en deux parties une partie contenant les images d'apprentissage et l'autre les images de test.

	Nombre d'exemples		Résultats
	Apprentissage	Test	
1	2	8	100%
2	3	8	80%
3	4	8	100%
4	5	8	80%
5	6	8	80%
6	7	8	100%
7	8	8	90%
8	9	8	90%

Tabl IV.1: Résultats de test.

IV.5. Conclusion

Au long de ce chapitre, j'ai vu les outils nécessaires pour la réalisation de mon application. J'ai cité l'environnement de développement. De plus, j'ai détaillé l'architecture : interfaces et testes de notre application.

Conclusion générale

Malgré les efforts et les travaux intensifs réalisés dans le domaine de la classification d'images, aucune méthode n'est jugé fiable à 100%, Mais au fur et à mesure les autres essayent d'améliorer les scores pour de meilleurs résultats.

Pour mon travail -classification des feuilles des plantes- j'ai choisi un descripteur global pour utiliser les invariants de Hu comme une méthode d'extraction des caractéristique; ses moments ce choix est justifié par leur efficacité dans les problèmes de classification, et j'ai choisi la méthode de carte auto-organisatrice de KOHONEN, comme méthode de classification, ce choix est justifié par leur principal avantage "*apprentissage*" qui permet de développer des systèmes dynamiques et évolutifs.

Le résultat obtenu est une application dynamique à la puissance de raisonnement de nom des classe ou de types des feuilles. Cette application peut être intégré ou utilisé avec des machines qui possède un environnement d'acquisition des images. De plus ce travail peut être utilise dans des autre domaines.

En fin ce travail reste ouvert pour des travaux de comparaison et/ou d'hybridation avec d'autres méthodes de classification (SVM, méthodes neuronales.....) et d'autres types de descripteurs (descripteurs de Fourier, ondelette).

Bibliographie

1. [ATE, 00] Atef Bel Hadj Ali, Mesures entre objets surfaciques. Application à la qualification des liens d'appariement, article, 2000.
2. [BED, 11] BEDOUHENE Saida, recherche d'images par le contenu, Mémoire de magister, Université mouloud Mammeri, Tizi-Ouzou, 2011.
3. [BEL, 08] BELHASSEN Mohamed, Une Approche d'Indexation et de Recherche d'Images Pulmonaires TDM par le Contenu Visuel, mémoire master, Université de MONASTIR, 2008.
4. [BES, 00] Bessai F.Z., Krelifaoui Mohamed, Guergab Mohamed, Utilisation des cartes auto-organisatrices de Kohonen dans la recherche documentaire, article, 2000.
5. [BOU, 09] BOUCHER Alain, Un outil pour l'indexation des vidéos personnelles par le contenu, Thèse de Doctorat, Université de Québec à trois-rivières, 2009.
6. [GRA, 98] Grady Booch, James Rumbaugh, Unified Modeling Language User Guide, Article, 1998.
7. [GUE, 07] Gueguen. L, Extraction d'information et compression conjointes des séries temporelles d'images satellitaires, Ecole Nationale Supérieure des Télécommunications Paris. Thèse de Doctorat, Octobre 2007.
8. [HAM, 03] HAMIDANE Rabie, BEKKARI fouad, La classification des types cellulaires par la carte de kohonen, mémoire ingénieur informatique, Université mohamed khider biskra, 2003
9. [HIL, 09] HILALI Hassane, Application de la classification textuelle pour l'extraction des règles d'association maximales, mémoire magister, Université QUÉBEC, 04/2009.
10. [HUG, 98] Hugues Bélanger, Réseau de Kohonen pour la détection des contours d'objets dans une image à niveaux de gris, projet d'application, Université QUEBEC, 1998.

11. [JAC, 01] Jacky Desachy, Analyse D'images, notes de cours - version 1.3, Université des Antilles et de la Guyane, 2001
12. [JIA, 01] Jianguo Zhang, Tieniu Tan, Brief review of invariant texture analysis methods, article, 2001.
13. [LAA, 14] Laalmi Houria, Bassou Oumessàd, Classification Des Feuilles de Plantes à Base de Moment de Hu, Mémoire de master, Université KASDI MARBAH OUARGLA, 06/2014.
14. [MES, 09] MESKALDJI Khouloud, Extraction et traitement de l'information: Un prototype d'un système de recherche d'images couleurs par le contenu, Université Mentouri d Constantine, 2009.
15. [MME, 12] Mme Benyahia Souad Née Belaidi, Application de la classe des méthodes d'apprentissage statistique SVM(support vector machine) pour la reconnaissance des formes dans les images, Mémoire de magister, Université Abou Bakr Belkaid– Tlemcen, 2012.
16. [MOU, 07] Mounib Noura, une approche Co-Evolutionnaire Proie-Prédateur pour la rehaussement d'images, Mémoire de Magister, Université colonel Hadj Lakhdar-Batna, 09/07/2007
17. [RAB, 11] RABIA Nacim, Revue des méthodes de segmentation d'images texturées: cas des images couleur, Mémoire de magister, Université mouloud Mammeri, Tizi-Ouzou, 30/06/2011.
18. [SOU, 14] Soufiane Khedairia, Contribution à la classification non supervisée: application aux données environnementales, mémoire doctorat, Université BADJI Mokhtar - Annaba, 2013-2014.
19. [TOU, 12] TOURQUI Wafa, Recherche d'images par le contenu symbolique (approche statistique) , Mémoire master, Université d'El-Oued, 2012.
20. [VUN, 11] VU NGOV Son, Indexation et recherche d'images par le contenu , Travail d'Intérêt Personnel Encadré (TIPE), Rapport final, Superviseur, 2011.

Site Web

21. [Web, 01] <http://www.map.toulouse.archi.fr/works/panoformation/imagenum/imagenum.htm>
(Consulté le 10/03/2015)
22. [Web, 02] [http://fr.wikipedia.org/wiki/Histogramme_\(imagerie_num%C3%A9rique\)#D.C3.A9finition](http://fr.wikipedia.org/wiki/Histogramme_(imagerie_num%C3%A9rique)#D.C3.A9finition)
(Consulté le 25/03/2015)
23. [Web, 03] http://fr.wikipedia.org/wiki/R%C3%A9seau_de_neurones_artificiels
(Consulté le 04/04/2015)
24. [Web, 04] http://fr.wikipedia.org/wiki/Carte_auto_adaptative
(Consulté le 04/04/2015)
25. [Web, 05] <http://www.ai-junkie.com/ann/som/som3.html>
(Consulté le 06/04/2015)
26. [Web, 06] <http://fr.wikipedia.org/wiki/NetBeans>
(Consulté le 04/05/2015)
27. [Web, 07] [http://fr.wikipedia.org/wiki/Java_\(langage\)](http://fr.wikipedia.org/wiki/Java_(langage))
(Consulté le 04/05/2015)
28. [Web, 08] http://fr.wikipedia.org/wiki/Extensible_Markup_Language
(Consulté le 05/05/2015)

Glossaire

C

CDDL :Common Development and Distribution License

G

GPLv2:The GNU General Public License version 2;

H

HTML: Hypertext Markup Language;

I

IDE: integrated development environment;

U

UML: Unified Modeling Language;

X

XML: Extensible Markup Language;

XHTML: Extensible Hyper Text Markup Language

XSLT: Extensible Stylesheet Language Transformations;

P

PHP: Hypertext Preprocessor

R

RGB : Red Green Blue: ;

RSS: Rich Site Summary;

S

SOM: self-organizing map;

SGML: Standard Generalized Markup Language;

SVG: Scalable Vector Graphics;

Résumé: L'objet de de cette étude est d'illustrer l'intérêt de l'utilisation des cartes d'auto-organisation dans un but de classification des images numériques. Cette méthode de classification construite à partir de l'algorithme de Kohonen dont l'apprentissage est non supervisé.

Dans ce contexte j'ai basé sur la description globale des images afin d'utiliser les moments invariants de Hu comme des entrées à la méthode de cartes d'auto-organisation. Après l'extraction des caractéristiques, une phase d'apprentissage sera effectuer.

Mot clé: Analyse des images, classification, moment Hu, catre auto-organisatrice de KOHONEN,

classification of plants with leaves map of Kohonen

Summary: The object of this study was to illustrate the possible uses of self-organizing maps (som) in classification of digital images. This classification method, based on the unsupervised learning algorithm of Kohonen.

In this context we have based on the global description of the images to use Hu moments invariants as inputs to method of self-organizing maps. After extracting the characteristics, a learning phase is performed.

Key words: Image analysis, classification, moment Hu, self-organizing catre of Kohonen.

تصنيف اوراق النباتات باستخدام خرائط كوهونان ذاتية التنظيم

ملخص: الغرض من هذه الدراسة هو توضيح الفائدة من استخدام خرائط ذاتية التنظيم في تصنيف الصور الرقمية. هذا الأسلوب من التصنيف بني على اساس خوارزمية (Kohonen) التي تعتمد على مبدأ التعلم غير المراقب. في هذا السياق اعتمدنا على الوصف الشامل للصور وذلك بغرض استخدام ثوابت (Hu) كمدخلات لطريقة خرائط التنظيم الذاتي. بعد استخراج الخصائص، يتم تنفيذ مرحلة التعليم.

الكلمات الدالة: تحليل الصور، التصنيف، ثوابت (Hu)، خرائط ذاتية التنظيم.