



République Algérienne Démocratique et Populaire  
Université Abou Bakr Belkaid- Tlemcen  
Faculté des Sciences  
Département d'Informatique



Mémoire de fin d'études

Pour l'obtention du diplôme de Master en Informatique

*Option: Modèle Intelligent et Décision(M.I.D)*

*Thème*

**Fusion de classifieurs supervisés pour la  
segmentation d'images microscopiques**

**Réalisé par :**

- BRAHIMI Abd elhamid

Présenté le 25 septembre 2011 devant le jury composé de MM.

- Mr hadjila Fethallah (Président)
- Mr Benazzouz Mortada (Encadreur)
- Mlle Baghli Ismahan (Co-Encadreur)
- Mr Merzoug Mohamed (Examineur)
- Mme Halfaoui A (Examineur)



Année universitaire : 2011-2012

Inscrit Sous le N°:  
Date le: 17 DEC. 2014  
Code: 166

Inscrit Sous le N°:  
Date le: 30 SEP. 2012  
Code: 1679

Je dédie ce travail à :

Mes parents,

Mes sœurs, mes frères

Mes amis,

Qu'ils trouvent ici l'expression de toute ma reconnaissance.

## *Remerciements*

*Nous devons toute chose à dieu le tout puissant et cela avant tout ; de nous avoir accordé la force, la chance et la santé pour avoir accompli ce modeste travail.*

*Ainsi à tous ceux et celles qui ont contribué directement ou indirectement à notre élaboration de ce mémoire.*

*Je tiens d'abord à exprimer mes profonds remerciements à Monsieur Mourtada BENAZZOUZ, en tant que directeur de mémoire de la confiance qu'elle m'a accordé dès le début de ce projet, ses conseils, son aide, sa présence, et son suivi durant mon projet ; grâce à son intérêt et sa compétence scientifique l'attention qu'il a porté sur mon travail et ces judicieux conseils.*

*Je tiens aussi à remercier Melle BAGHLI I, en tant que Co-encadreur, de la confiance, de son aide, elle a largement contribué à la diversité des travaux réalisés et de l'amélioration du manuscrite.*

*Je tiens à exprimer mes remerciements aux membres du jury, qui ont accepté d'évaluer mon travail. Merci au président Mr hadjila F., aux examinateurs Mr Marzoug M., Mme Halfaoui A.*

*Sans oublier nos amis et collègues qui nous ont épaulés durant ce long cursus.*

*Merci*

# Table des matières

Table des figures .....	4
Table des tableaux .....	5
Introduction générale .....	6
<b>Chapitre I : Traitement d'image et Espaces de couleur</b>	
I.1 Introduction .....	8
I.2 Image couleur .....	8
I.3 Espaces couleur .....	10
I.3.1 Système RGB .....	10
I.4.2 Système normalisé (r, g, b) .....	11
I.4.3 Système XYZ .....	11
I.4.4 Système L*u*v .....	12
I.4.5 Système L*a*b* .....	13
I.4.6 Le système HSL .....	14
I.4.7 Système HSV .....	15
I.4.8 L'espace YUV .....	16
I.4.9 L'espace $I_1I_2I_3$ .....	16
I.5 Traitement et analyse d'image .....	17
I.5.1 Segmentation .....	17
I.6 Conclusion .....	18
<b>Chapitre II : les méthodes de classification</b>	
II.1 Introduction .....	19
II.2 Séparateurs à Vaste Marge (SVM) .....	19
II.2.1 Principe de la technique SVM .....	20
II.2.2 Classifieur linéaire .....	20
II.2.3 Marge maximale de l'hyperplan .....	21

II.2.4 Minimisation quadratique sous contraintes .....	23
II.2.5 SVM non-linéaire .....	25
II.2.6 Relaxation des contraintes .....	26
II.2.7 SVM pour le cas multi-classes .....	27
II.2.8 Avantages et inconvénients .....	27
II.3 Réseaux de Neurones .....	28
II.3.2 Neurone biologique .....	28
II.3.3 Principe de fonctionnement .....	29
II.3.3.1 Neurone formel .....	29
II.3.3.2 Calcul des poids synaptiques .....	31
II.3.4 Quelques réseaux célèbres .....	31
II.3.4.1 Le perceptron .....	31
II.3.4.2 Les perceptrons multicouches (PMC) .....	31
II.3.4.3 Les réseaux de RBF (Radial Basis Fonction) .....	32
II.3.5 Avantages et inconvénients .....	33
II.4 K plus proches voisins .....	34
II.4.1 Principe du classifieur KPPV .....	34
II.4.2 Choix du nombre .....	35
II.4.3 Définition de la distance .....	35
II.4.4 Sélection de la classe .....	36
II.4.5 Avantages et inconvénients .....	37
II.5 Conclusion .....	37
<b>Chapitre III : Fusion d'information</b>	
III.1 Introduction .....	38
III.2 Typologies et niveaux de la fusion .....	38
III.3 Stratégies de fusion .....	40

III.4 Etapes de l'opération de fusion .....	41
III.5 Les méthodes de fusion .....	42
III.5.1 Principe du vote .....	42
III.5.2 Théorie des probabilités .....	44
III.5.3 Théorie des possibilistes .....	45
III.6 Conclusion .....	46
<b>Chapitre IV: Application</b>	
VI.1 Préliminaire .....	47
VI.2 Environnement de travail .....	47
VI.2.1 Base d'images .....	47
VI.2.2 outil utilisé .....	48
VI.3 Phase d'apprentissage .....	48
VI.3.1 Construction de la base d'apprentissage .....	49
VI.3.2 Construction et test des classifieurs .....	49
IV. 4 Phase de fusion .....	51
IV.4.1 Vote majoritaire .....	52
IV.4.2 Vote majoritaire pondéré .....	53
IV.5 Discussion des résultats .....	55
Conclusion générale et perspectives .....	58
Références bibliographiques .....	59

# Table des figures

Figure I.1 : Représentation d'un pixel dans un espace couleur .....	9
Figure I.2 : Nuage de points de l'image dans un espace couleur.....	9
Figure I.3 : Représentation du modèle « RGB ».....	10
Figure I.4 : Représentation du modèle « XYZ ».....	12
Figure I.5 : Représentation du modèle « $L^*a^*b^*$ ».....	14
Figure I.6 : Représentation du modèle «HSL».....	15
Figure I.7 : Représentation du modèle «HSV».....	16
Figure II.1 : hyperplan séparateur.....	21
Figure II.2 : Illustration de la marge maximale et des vecteurs supports .....	22
Figure II.3 : neurone biologique .....	29
Figure II.4 : neurone formel (artificiel) .....	30
Figure II.5 : schéma d'un perceptron multicouche .....	32
Figure II.6 : schéma d'un RBF .....	32
Figure II.7. Méthode de K-plus proches voisins.....	35
Figure III.1 : Une architecture de fusion de classifieurs en parallèle .....	41
Figure VI.1: Exemple d'image cytologique .....	47
Figure VI.2: Image étiquetée .....	48
Figure VI.3: Définition des 4 régions .....	49
Figure IV.4: Résultats obtenus pour les trois classifieurs .....	51
Figure IV.5: Stratégie de fusion .....	52
Figure IV.6: Interface du choix de l'image et de la segmentation.....	54
Figure IV.7: Affichage des détails du résultat .....	54
Figure IV.8: Premier résultat .....	56
Figure IV.9: Deuxième résultat .....	57

# Table des tableaux

TABLE IV.1 – Résultats obtenus pour le classifieur KPPV.....	49
TABLE IV.2 – Résultats obtenus pour le classifieur PMC.....	50
TABLE IV.3 – Résultats obtenus pour le classifieur SVM .....	50
TABLE IV.6 : Résultats obtenus pour le vote majoritaire.....	52
TABLE IV.7: Précisions obtenues sur la base d'apprentissage .....	53
TABLE IV.8 – Résultats obtenus pour le vote majoritaire pondéré.....	53



**Introduction générale :**

- **Problématique :** Les cyto-pathologistes ont une démarche diagnostique essentiellement basée sur l'analyse visuelle d'images microscopiques. Cette démarche repose sur la recherche d'anomalies décelables en raison des modifications de la structure des cellules qu'elles entraînent. La facilité et la fiabilité des diagnostics dépendent de l'expérience et de l'entraînement à la lecture microscopique, et surtout du degré d'attention et d'acuité porté à la recherche et à l'interprétation de signes de modifications cellulaires significatifs parfois très difficiles à reconnaître, d'où l'intérêt d'utiliser un système d'aide à la décision. Pour pallier cela, une approche consiste à aider le cytopathologiste par un système semi-automatique afin de confirmer son analyse visuelle (segmentation automatique). La segmentation d'image constitue une partie clé d'un tel système qui conditionne toutes les étapes ultérieures (caractérisation et reconnaissance pathologies ). La segmentation est un domaine réputée difficile en analyse d'images. En effet il n'existe pas de théories s'appliquant à différents types d'images, mais plutôt des méthodes variées que l'on choisit et que l'on développe pour résoudre des problèmes d'analyse sur un type d'image bien défini. Le choix de la méthode est généralement motivé par la représentation d'images et le contexte. En imagerie médicale microscopique, ce choix est d'autant plus difficile que l'on a à traiter différents types d'images selon les organes étudiés, la coloration adaptée, la variabilité des objets cellulaires et tissulaires observés.
- **Contribution :** Nous désirons élaborer une méthode de segmentation d'images couleur, en séparant ses composantes en globule blanc ou leucocyte (noyau et cytoplasme), globule rouge et fond. La qualité de cette segmentation est primordiale puisque la qualité du diagnostic en dépend. La couleur constituant une source d'information importante dans le domaine de la microscopie biomédicale cellulaire, nous présenterons une méthode de segmentation d'images par classification pixellaire basée sur la couleur. L'influence du choix porte sur la sélection des espaces couleur qui permettent de discriminer au mieux les classes de pixels en présence. Comme cette approche multi-espaces

## *Introduction générale*

couleur ne tient compte que des distributions des couleurs présentes dans les images, nous proposons une classification pixellaire supervisée réalisée avec les algorithmes de Machines à Supports de Vecteurs (SVM), K Plus Proches Voisins (KPPV), Perceptron Multi-Couches (PMC). L'utilisation de plusieurs classifications pixellaires entraînant des conflits de décisions (3 différents schémas de segmentation), ainsi nous proposons deux méthode de combinaison pour obtenir la segmentation finale.

- **Organisation du manuscrit :** Le travail que nous présentons dans ce mémoire est organisé en quatre chapitres :
  - ✓ Chapitre I : ce chapitre aborde la notion de couleur en présentant les différents espaces couleur qui la représente,
  - ✓ Chapitre II : Les différentes méthodes de classification supervisé utilisées,
  - ✓ Chapitre III : ce chapitre présente la fusion de décision, ainsi que les différentes étapes qui la constituent.
  - ✓ Chapitre IV : ce chapitre présente les expérimentations ainsi que les différentes discussions concernant les résultats obtenus.
  - ✓ Et nous terminons par la conclusion et les perspectives ouvertes par notre travail

# **Chapitre I :**

## **Traitement d'image et Espaces de couleur**

## **I.1 Introduction**

Durant ces dernières années, l'informatique s'est affirmée comme une nouvelle technologie majeure. Les progrès des moyens informatiques et des techniques de traitement ont permis d'ouvrir une voie de développement très prometteuse. L'amélioration et la multiplication des moyens de diffusion des données informatiques a pris également une grande part à la multiplication de ces techniques. Les outils de communication utilisent de plus en plus les images, ce qui a entraîné un accroissement des recherches dans les domaines du traitement et de l'analyse d'images. Ces techniques ont été développées avec succès dans des domaines divers et très variés (images industrielles, de robotique, biologiques, médicales, aériennes, etc.).

Dans ce chapitre, nous commencerons par présenter la nature des images ainsi que les protocoles d'acquisition. Nous présenterons ensuite quelques modèles de représentation de la couleur. Nous définirons ensuite les principaux espaces couleur que nous pouvons trouver dans la littérature. Après, nous passerons au domaine du traitement et d'analyse de l'image, plus particulièrement à la segmentation des images.

## **I.2 Image couleur**

Informatiquement parlé, une image couleur est avant tout une image numérique. Cette dernière est un tableau (une matrice) de dimensions finies de valeurs numériques de précision finie. Chaque élément fondamental de ce tableau s'appelle un pixel (de l'anglais Picture élément). Chaque pixel est caractérisé par sa position spatiale dans la matrice et sa valeur numérique.

Une image est acquise à l'aide d'un capteur, et selon la capacité spectrale de ce dernier, trois classes de capteurs peuvent être différenciées : panchromatique, multi spectrale et hyper-spectrale.

Les senseurs panchromatiques sont sensibles à l'ensemble du spectre visible et l'information est contenue à l'intérieur d'une seule large bande spectrale. Les senseurs multi spectraux sont sensibles à certaines bandes prédéfinies

du spectre électromagnétique dans le visible et dans l'infrarouge alors que les senseurs hyper spectraux sont sensibles à une large bande du spectre électromagnétique (prédéfinies), divisée en plusieurs petites bandes spectrales.

Dans le cas général [Porebski06], la couleur d'un pixel va être représentée par trois composantes notées  $C_1$ ,  $C_2$  et  $C_3$ , et à ces trois composantes nous faisons correspondre respectivement l'origine  $O$  appelé espace couleur. Dans cet espace, chaque couleur est ainsi représenté par un point  $C$ , qui définit le vecteur couleur  $\overrightarrow{OC}$  et dont les coordonnées sont les valeurs des composantes  $C_1$ ,  $C_2$  et  $C_3$ , la figure I.1 illustre ce propos. Une image sera donc représentée. Dans l'espace couleur ( $C_1$ ,  $C_2$ ,  $C_3$ ) par un nuage de points (figure I.2).

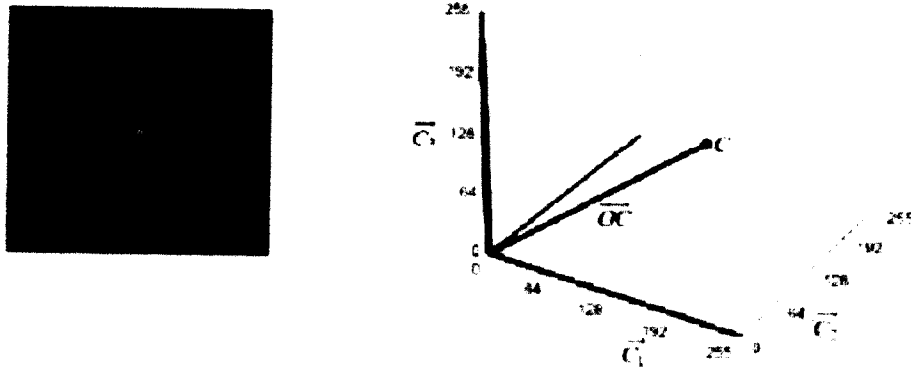


Figure I.1 : Représentation d'un pixel dans un espace couleur

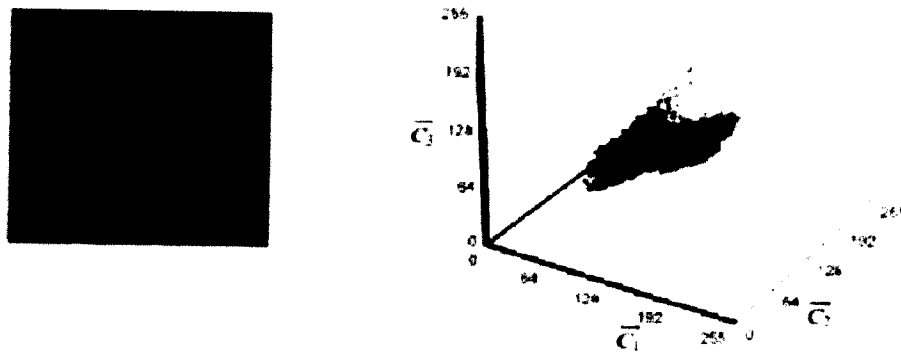


Figure I.2 : Nuage de points de l'image dans un espace couleur

### I.3 Espaces couleur

D'après le principe du tri variance visuelle, toute couleur peut être reproduite visuellement à l'identique dans des conditions d'observation déterminées par le mélange algébrique en proportions définies de manière unique de trois couleurs, appelées primaires [Gagliari97, Sclaroff97]. Ces trois couleurs peuvent être arbitrairement choisies sous la seule réserve qu'aucune d'entre elles ne puisse être reproduite par un mélange approprié des deux autres. Ce principe est à l'origine de tous les espaces de représentation de la couleur de ce fait chaque espace présente des caractéristiques particulières [Striker95, Gonzales02], nous citerons :

#### I.3.1 Système RGB

Le système le plus couramment utilisé est le système RGB (Red-Green-Blue), qui est le système des trois couleurs fondamentales. Il associe à chaque couleur trois composantes (ou canaux), qui correspondent aux intensités respectives de trois couleurs primaires de la synthèse additive. Le blanc correspond à la valeur maximale pour chaque canal, tandis que le noir correspond aux trois composantes nulles.

En pratique, les valeurs de chaque canal sont des entiers compris entre 0 et  $N_R$  pour le canal rouge, entre 0 et  $N_G$  pour le canal vert et entre 0 et  $N_B$  pour le canal bleu. Ainsi, chaque couleur appartient à un parallélépipède (Figure I.3). Le codage le plus couramment utilisé consiste à prendre  $N_R = N_G = N_B = 255$ , ce qui permet de stocker chaque composante sur un octet.

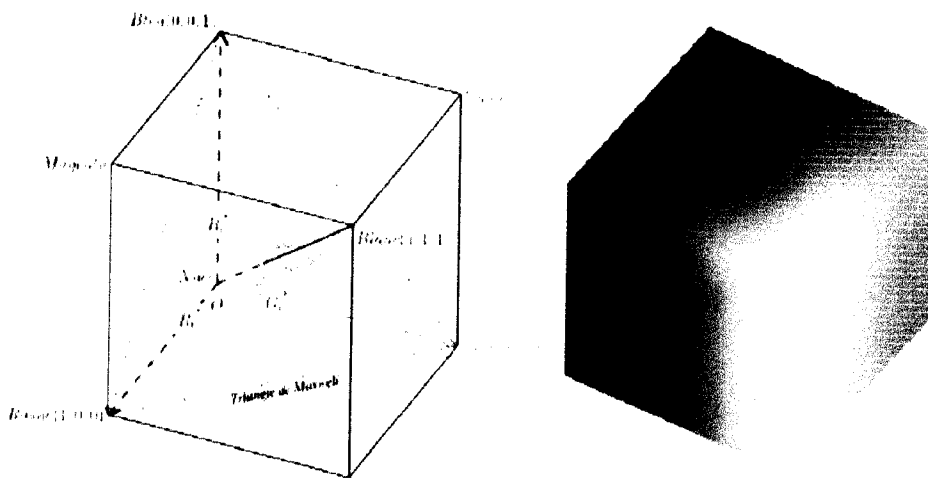


Figure I.3 : Représentation du modèle « RGB »

#### I.4.2 Système normalisé (r, g, b)

Un des problèmes du système RGB est que trois canaux ne représentent pas seulement la couleur, mais aussi la luminosité. Le Système (r, g, b) permet de supprimer en partie en normalisant les trois composantes R, G, et B. On obtient alors trois canaux r, g et b.

Le passage du système RGB au système normalisé (r, g, b) s'effectue en divisant chaque composante par la somme des trois :

$$\begin{cases} r = \frac{R}{R+G+B} \\ g = \frac{G}{R+G+B} \\ b = \frac{B}{R+G+B} \end{cases} \quad (1.2)$$

Ainsi, la somme des trois nouvelles composantes est égale à 1. Chaque couleur normalisée est un réel appartenant à l'intervalle.

#### I.4.3 Système XYZ

Le système XYZ a été établi par la CIE<sup>1</sup> afin de pallier à certains inconvénients du système RGB. Ce système correspond à un changement de couleurs primaires et s'obtient simplement à partir du système RGB à l'aide d'une matrice de passage. Les coefficients de cette matrice sont déterminés par rapport à un blanc de référence que l'on appelle illuminant. Celui que nous utilisons est appelé illuminant standard D65. Qui est représenté par les variables des illuminant  $x$  et  $y$  (Figure I.4).

Ce système de couleurs ne nous intéresse pas particulièrement, mais il constitue une étape intermédiaire pour passer au système  $L^*u^*v^*$ .

Le passage du système RGB que nous utilisons (tous les canaux sont codés par des entiers entre 0 et 255) au système XYZ se fait par le calcul matriciel suivant :

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} 0.430574 & 0.341550 & 0.178325 \\ 0.222015 & 0.706655 & 0.071330 \\ 0.020183 & 0.129553 & 0.939180 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix} \quad (1.2)$$

<sup>1</sup> CIE (Commission Internationale de l'Éclairage) est une organisation internationale chargée d'établir des normes et des recommandations reconnues par tous les pays, afin de pouvoir quantifier la couleur.

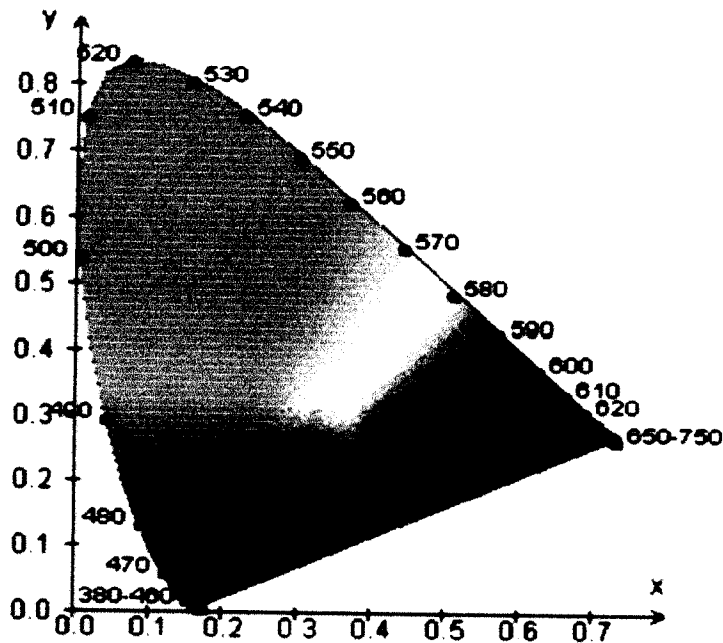


Figure I.4 : Représentation du modèle « XYZ »

#### I.4.4 Système $L^*u^*v^*$

Le problème avec l'espace  $XYZ$  est son manque d'uniformité dans la perception, à savoir que deux couleurs à la même distance d'une autre ne sont pas nécessairement perpétuellement équidistantes [Healey96]. Pour corriger cet inconvénient la CIE a proposé en 1976 le modèle  $L^*u^*v^*$  qui est perceptuellement uniforme, c'est-à-dire que si deux couleurs  $A$  et  $B$  sont aussi proches entre elles que deux autres couleurs  $C$  et  $D$ , alors la différence perçue entre  $A$  et  $B$  sera équivalente à celle perçue entre  $C$  et  $D$ . Les équations permettant de passer du système  $XYZ$  au  $L^*u^*v^*$  sont [Tremeau93]:

La luminance  $L^*$  se calcule par :

$$L^* = \begin{cases} 116 \sqrt[3]{\frac{Y}{Y_0}} - 16 & \text{si } \frac{Y}{Y_0} > 0.008856 \\ 903.3 \frac{Y}{Y_0} & \text{si } \frac{Y}{Y_0} \leq 0.008856 \end{cases} \quad (1.3)$$

Les autres

$$u^* = 13L^*(u' - u'_0) \quad (1.4)$$

$$v^* = 13L^*(v' - v'_0) \quad (1.5)$$



Avec :

$$u' = \frac{4X}{X+15Y+3Z} \quad (1.6)$$

$$v' = \frac{9X}{X+15Y+3Z} \quad (1.7)$$

$$u'_0 = \frac{4X_0}{X_0+15Y_0+3Z_0} \quad (1.8)$$

$$v'_0 = \frac{9X_0}{X_0+15Y_0+3Z_0} \quad (1.9)$$

Où  $X_0, Y_0, Z_0$  sont les composantes d'un blanc de référence dans l'espace XYZ.  $L^*$  est la luminance et  $u^*$  et  $v^*$  donnent la chrominance de la couleur.

#### I.4.5 Système $L^*a^*b^*$

Le système  $L^*a^*b^*$  est comme le système  $L^*u^*v^*$  perceptuellement uniforme. La différence entre ces deux espaces de couleur vient de l'information sur la chrominance. La formule pour obtenir la valeur de la luminance  $L^*$  est la même, elle est donnée dans l'équation précédente, les composantes de chrominance sont donnée par :

$$a^* = 500 \left( f\left(\frac{x}{x_0}\right) - f\left(\frac{y}{y_0}\right) \right) \quad (1.10)$$

$$b^* = 200 \left( f\left(\frac{y}{y_0}\right) - f\left(\frac{z}{z_0}\right) \right) \quad (1.11)$$

Avec  $f$  défini par :

$$f(x) = \begin{cases} \sqrt[3]{x} & \text{si } x > 0.008856 \\ 7.787x + \frac{16}{116} & \text{si } x \leq 0.008856 \end{cases} \quad (1.12)$$

Comme le système  $L^*u^*v^*$ , ce système est perceptuellement uniforme. Sa représentation spatiale est donnée par une sphère (Figure I.5).

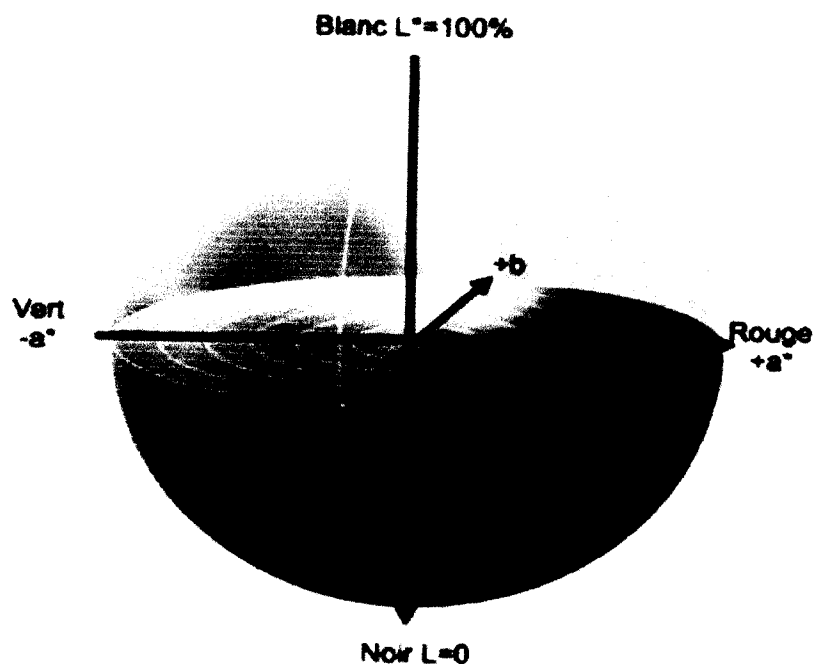


Figure I.5 : Représentation du modèle « $L^*a^*b^*$ »

#### I.4.6 Le système HSL

Cet espace est basé sur le modèle de Pratt. Comme son nom l'indique il est basé sur les trois grandeurs de Teinte, Saturation et Luminance. La Figure 11 donne une représentation de cet espace (dans l'espace euclidien  $RGB$ ). Cet espace est obtenu à partir de l'espace  $RGB$  et est décrit par les équations suivantes [Kunt 91] :

$$H = a \cos \left[ \frac{(R-G)+(R-B)}{\sqrt{(R-G)^2+(R+B)(G-B)}} \right] \text{ et si } G < B \text{ } H = 2\pi - H \quad (1.13)$$

$$S = 1 - \frac{3\min(R,G,B)}{L} \quad (1.14)$$

$$L = R + G + B \quad (1.15)$$

Le non linéarité des deux équations permettant de calculer  $H$  et  $S$  entraîne des discontinuités dans leurs valeurs et deux couleurs perceptuellement proches peuvent se transformer en des composantes de teinte et de saturation très différentes. Ceci est dû aux singularités de la teinte pour les points achromatiques ( $R=G=B$ ) et de la saturation pour le noir ( $R=G=B=0$ ).

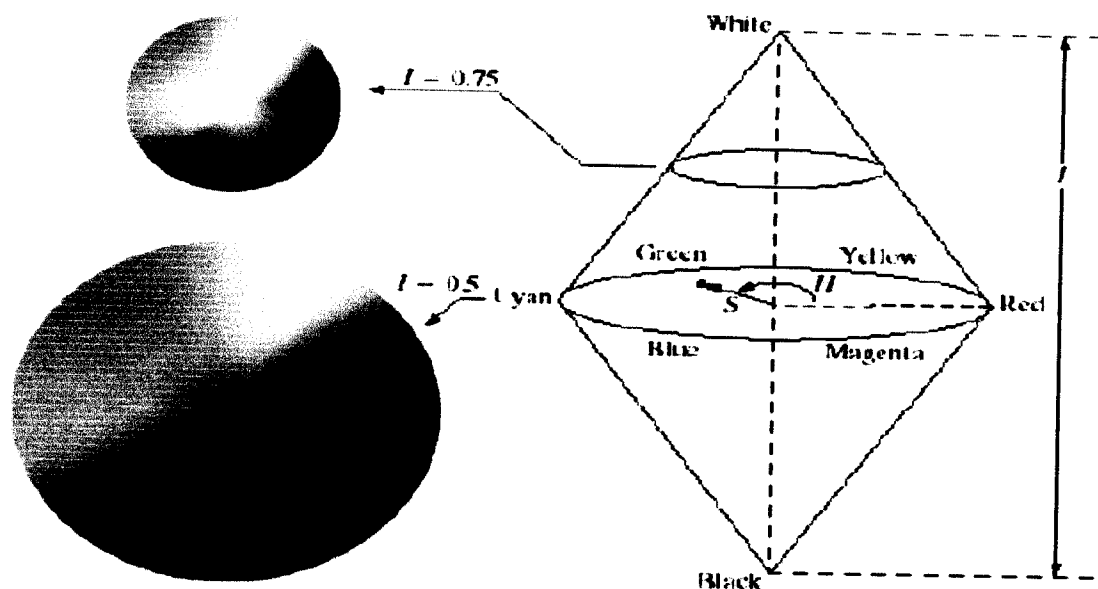


Figure I.6 : Représentation du modèle «HSL»

#### I.4.7 Système HSV

Le système (Hue – Saturation – Value) est défini par un cylindre qui représente la teinte, la saturation et la valeur d'une couleur. La teinte H est représentée par un angle entre 0 et 360° : elle indique la famille de couleur (rouge, jaune, vert, bleu, etc.). La saturation S donne une information sur la pureté de la couleur. La valeur V correspond à l'intensité lumineuse. Elle indique si la couleur est claire ou sombre. L'algorithme de transformation du système RGB vers HSV est donné par :

$$V = \max(R, G, B) \quad (1.16)$$

$$S = \frac{255(V - \min(R, G, B))}{V} \quad \text{si } V \neq 0, 0 \text{ sinon} \quad (1.17)$$

$$H = \begin{cases} \frac{60(G-B)}{S} & \text{si } V = R \\ 180 + \frac{60(B-R)}{S} & \text{si } V = G \\ 124 + \frac{60(R-G)}{S} & \text{si } V = B \end{cases} \quad (1.18)$$

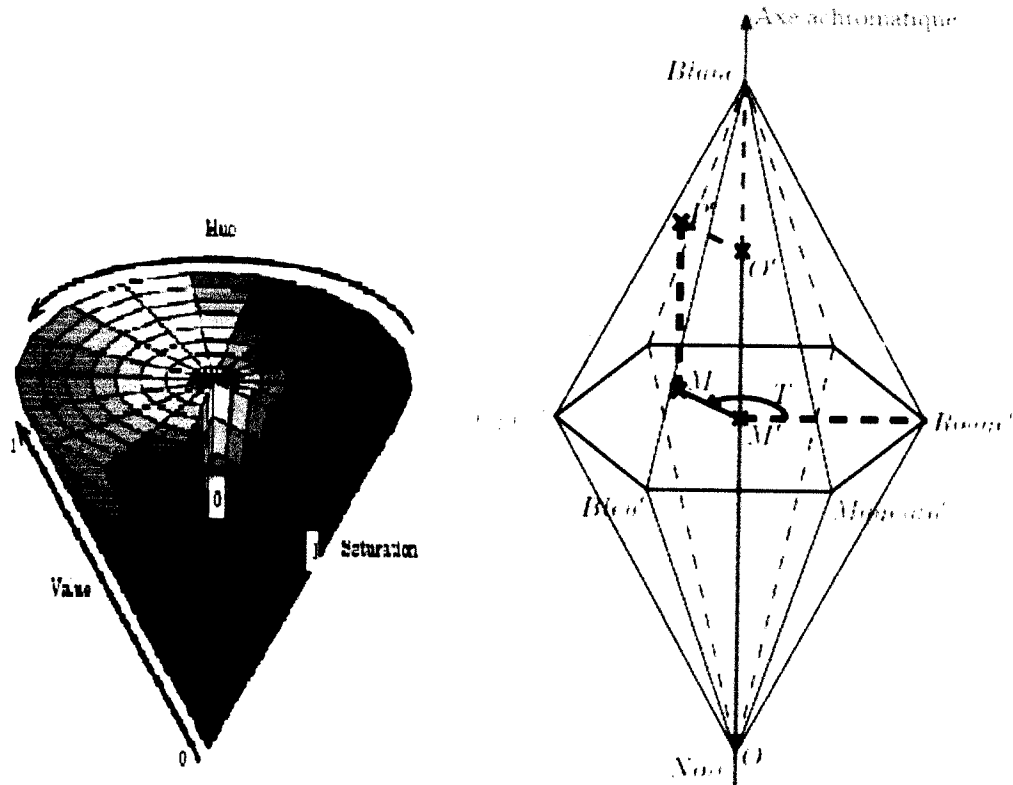


Figure I.7 : Représentation du modèle «HSV»

#### 1.4.8 L'espace YUV

Le modèle YUV est un modèle de représentation de la couleur dédié à la vidéo analogique et utilisé dans les standards vidéo PAL (Phase Alternation by Line) et SECAM (SEquentiel Couleur A Mémoire). Il utilise l'illuminant D65 comme blanc de référence. L'espace YUV présent dans les travaux de ULTRÉ, LIANG, CLARAMONT [ULTRÉ96, LIANG87, CLARAM84] et dont la transformation est issue de l'espace RGB est définie comme suit :

$$\begin{pmatrix} Y \\ U \\ V \end{pmatrix} = \begin{pmatrix} 0.299 & 0.587 & 0.114 \\ -0.147 & -0.289 & -0.436 \\ 0.615 & -0.515 & -0.100 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix} \quad (1.19)$$

Où Y représentent la luminance, U et V la chrominance.

#### 1.4.9 L'espace $I_1I_2I_3$

Il est basé sur le modèle du tri-stimulus. Il a été obtenu par Ohta [Ohta 80] au moyen d'une transformation de Karhunen-Loève. Celle-ci est un outil statistique qui permet de trouver pour chaque image le système de coordonnées le plus discriminant pour celle-ci. Les composantes  $X_1, X_2, X_3$  calculées ainsi sont :

$$X_i = W_{R_i}R + W_{V_i}G + W_{B_i}B \quad (1.20)$$

Où  $W_i = (W_{R_i}, W_{V_i}, W_{B_i})^T$  est le vecteur propre normalisé de la matrice de covariance des distributions de  $R, G, B$  correspondant à la valeur propre  $\lambda_i$  avec  $\lambda_1 \geq \lambda_2 \geq \lambda_3$ . Les composantes obtenues sont non corrélées,  $X_1$  est le vecteur de base optimal (ayant la plus grande variance). Les composantes  $I_1 I_2 I_3$  sont donc importantes pour représenter l'information couleur. L'espace  $I_1 I_2 I_3$  s'obtient à partir de l'espace  $RGB$  par :

$$I_1 = \frac{(R+G+B)}{3} \quad I_2 = \frac{(R-B)}{2} \quad I_3 = \frac{(2G-R-B)}{3} \quad (1.21)$$

## 1.5 Traitement et analyse d'image

Le domaine qui regroupe l'ensemble des méthodes et techniques opérant sur les images est le traitement et analyse d'images. Ce dernier s'intéresse, entre autres, à l'amélioration de l'aspect visuel de l'image (filtrage, restauration,...) et à l'extraction des informations jugées pertinentes (Extraction de primitives et segmentation).

Dans cette section, nous donnerons un aperçu sur le domaine de la segmentation d'image.

### 1.5.1 Segmentation

Fondamentalement, la segmentation est un processus qui consiste à découper une image en régions connexes présentant une homogénéité selon un certain critère, comme par exemple la couleur. L'union de ces régions doit redonner l'image initiale [Cocquerez95].

La segmentation est une étape importante pour l'extraction des informations qualitatives de l'image.

On regroupe généralement les algorithmes de segmentation en trois grandes classes [Porebski06] :

- **Segmentation basée sur les régions** : Elle correspond aux algorithmes d'accroissement ou de découpage de région. Dans l'accroissement de région on part d'un ensemble de petites régions uniformes dans l'image (de la taille d'un ou de quelques pixels) et on regroupe les régions adjacentes de même couleur jusqu'à ce qu'aucun regroupement ne soit plus possible.

Pour le découpage de région, on part de l'image entière que l'on va subdiviser récursivement en plus petites régions tant que ces régions ne seront pas suffisamment homogènes.

Les algorithmes dit split and merge : Sont un mélange de ces deux approches.

- **Segmentation basée sur les contours** : Cette catégorie correspond à une approche duale qui cherche à détecter une transition(ou contour) entre deux régions connexes. Les détecteurs de contours peuvent être simples, comme les operateurs de Sobel ou de Roberts, ou plus complexes tel que l'opérateur de Canny. La segmentation par approche contour peut être faite à partir de l'image de gradient, du laplacien par détection des contours ou par application d'un contour déformable initialisé automatiquement ou manuellement.
- **Segmentation basée sur les pixels**: Le principe consiste à regrouper les pixels selon leurs attributs sans tenir compte de leur localisation au sein de l'image. Cela permet de construire des classes de pixels ; les pixels adjacents, appartenant à une même classe, forment alors des régions. Parmi les approches pixellaires nous distinguons les méthodes de seuillage et les méthodes de classification (clustering).

## **I.6 Conclusion**

Dans ce chapitre, nous avons abordé des généralités sur les principaux espaces couleurs et rappelé la définition de la segmentation d'une image. Nous avons présenté rapidement quelques grandes familles de méthodes de la segmentation.

Dans le chapitre suivant, Nous allons étudier les différentes méthodes de classification et les différentes manières de combiner ces méthodes.

# **Chapitre II :**

## **Les méthodes de classification**

## II.1 Introduction

Les méthodes de classification ont pour but d'identifier les classes auxquelles appartiennent des objets à partir de certains paramètres descriptifs. Elles s'appliquent à un grand nombre d'activités humaines et conviennent en particulier au problème de la prise de décision automatisée. La procédure de classification sera extraite automatiquement à partir d'un ensemble d'exemples. Un exemple consiste en la description d'un cas avec la classification correspondante. Un système d'apprentissage doit alors, à partir de cet ensemble d'exemples, extraire une procédure de classification; il s'agit en effet d'extraire une règle générale à partir des données observées. La procédure générée devra classer correctement les exemples de l'échantillon et avoir un bon pouvoir prédictif pour classer correctement de nouvelles descriptions.

Les méthodes utilisées pour la classification sont nombreuses, nous présenterons dans la suite de ce chapitre une étude détaillée des trois techniques: Séparateurs à Vaste Marge (SVM), réseaux de neurones (RN) et K plus proches voisins (KPPV); puisque ces dernières ont été choisies comme méthodes de classification des pixels dans nos expérimentations.

## II.2 Séparateurs à Vaste Marge (SVM)

Les machines à vecteurs de support (Support Vector Machine, SVM) appelés aussi séparateurs à vaste marge sont des techniques d'apprentissage supervisées destinées à résoudre des problèmes de classification. Les machines à vecteurs supports exploitent les concepts relatifs à la théorie de l'apprentissage statistique et à la théorie des bornes de Vapnik et Chervonenkis [Vapnik95]. La justification intuitive de cette méthode d'apprentissage est la suivante: si l'échantillon d'apprentissage est linéairement séparable, il semble naturel de séparer parfaitement les éléments des deux classes de telle sorte qu'ils soient le plus loin possibles de la frontière choisie. Ces fameuses machines ont été inventées en 1992 par Boser et al. [Cortes95], mais leur dénomination par SVM n'est apparue qu'en 1995 avec Cortes et al. [Boser92]. Depuis lors, de nombreux développements ont été réalisés pour proposer des variantes traitant le cas non-linéaire ([Burges98, Smola98]). Le succès de cette méthode est justifié par les solides bases théoriques qui la soutiennent. Elles permettent d'aborder des problèmes très divers dont la



classification. SVM est une méthode particulièrement bien adaptée pour traiter des données de très haute dimension.

### II.2.1 Principe de la technique SVM

Cette technique est une méthode de classification à deux classes qui tente de séparer les exemples positifs des exemples négatifs dans l'ensemble des exemples. La méthode cherche alors l'hyperplan qui sépare les exemples positifs des exemples négatifs, en garantissant que la marge entre le plus proche des positifs et des négatifs soit maximale.

Cela garantit une généralisation du principe car de nouveaux exemples pourront ne pas être trop similaires à ceux utilisés pour trouver l'hyperplan mais être situés d'un côté ou l'autre de la frontière. L'intérêt de cette méthode est la sélection de vecteurs supports qui représentent les vecteurs discriminant grâce auxquels est déterminé l'hyperplan. Les exemples utilisés lors de la recherche de l'hyperplan ne sont alors plus utiles et seuls ces vecteurs supports sont utilisés pour classer un nouveau cas, ce qui peut être considéré comme un avantage pour cette méthode.

### II.2.2 Classifieur linéaire

Un classifieur est dit linéaire lorsqu'il est possible d'exprimer sa fonction de décision par une fonction linéaire en  $x$ . On peut exprimer une telle fonction par:

$$h(x) = \langle w, x \rangle + b = \sum_{i=1..n} w_i x_i \quad (2.1)$$

où  $w \in \mathbb{R}^n$  est le vecteur de poids et  $b \in \mathbb{R}^0$  le biais, alors que  $x$  est la variable du problème.  $X$  est l'espace d'entrée et qui correspond à  $\mathbb{R}^n$  où  $n$  est le nombre de composantes des vecteurs contenant les données. Notons que l'opérateur  $\langle \rangle$  désigne le produit scalaire usuel dans  $\mathbb{R}^n$ ,  $w$  et  $b$  sont les paramètres à estimer de la fonction de décision  $h(x)$ .

Pour décider à quelle catégorie un exemple estimé  $x'$  appartient, il suffit de prendre le signe de la fonction de décision  $y = \text{sign}(h(x'))$ . La fonction est appelée  $\text{sign}()$  classifieur. Géométriquement (voir figure II.1), cela revient à considérer un hyperplan qui est le lieu des points  $x$  satisfaisant  $\langle w, x \rangle + b = 0$ . En orientant l'hyperplan, la règle de décision correspond à observer de quel côté de l'hyperplan se trouve l'exemple  $x'$ . On voit que le vecteur  $w$  définit la pente de l'hyperplan ( $w$  est perpendiculaire à l'hyperplan). Le terme  $b$  quant à lui permet de translater l'hyperplan parallèlement à lui-même.

L'objectif de la discrimination linéaire est de trouver la bonne fonction de décision  $h(x)$ . La classe de tous les hyperplans qui en découle sera notée  $H$ .

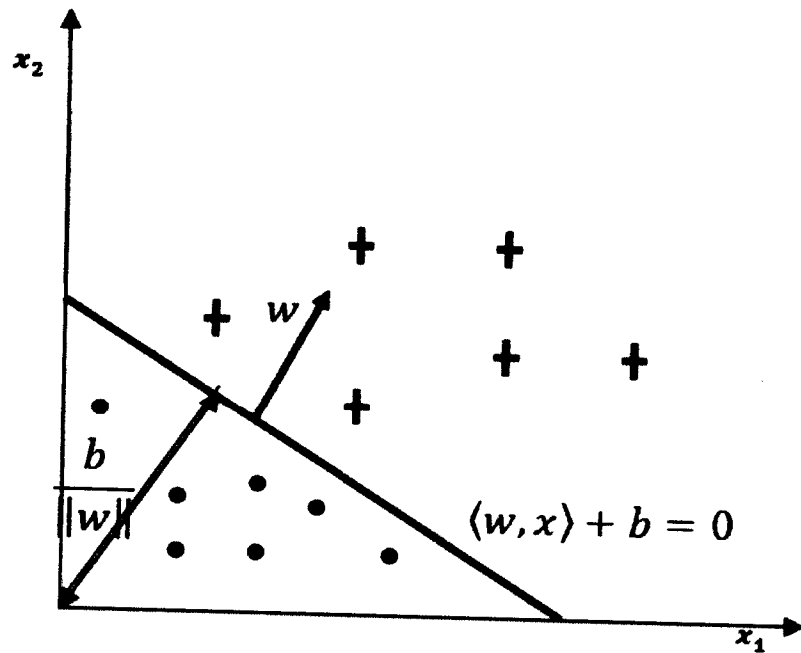


Figure II.1 : hyperplan séparateur

### II.2.3 Marge maximale de l'hyperplan

La marge est la distance entre la frontière de séparation et les échantillons les plus proches. Dans les SVM, la frontière de séparation est choisie comme celle qui maximise la marge.

La marge géométrique représente la distance euclidienne prise perpendiculairement entre l'hyperplan et l'exemple  $x_i$ . En prenant un point quelconque  $x_p$  se trouvant sur l'hyperplan, la marge géométrique peut s'exprimer par :

$$\frac{w}{\|w\|} (x_i - x_p) \quad (2.2)$$

L'hyperplan à marge maximale est le modèle le plus utilisé dans les machines à vecteurs supports. L'estimation des paramètres  $(w^*, b^*)$  de l'hyperplan qui maximise la marge se fait en résolvant le problème d'optimisation suivant :

$$(w^*, b^*) = \operatorname{argmax}_{(w,b)} \{ \min_i (y_i (wx_i + b)), \|w\| = 1 \} \quad (2.3)$$

Dire que les deux classes de l'échantillon d'apprentissage  $S$  sont linéairement séparables est équivalent à dire qu'il existe des paramètres  $(w^*, b^*)$  tels que l'on a pour tout  $i (= 1, 2, \dots, n)$  :

$$w^* x_i + b^* > 0 \text{ si } y_i = 1 \quad (2.4)$$

$$w^* x_i + b^* < 0 \text{ si } y_i = -1 \quad (2.5)$$

ce qui est équivalent à :

$$y_i(w^* x_i + b^*) > 0; \forall i = 1, 2, \dots, n \quad (2.6)$$

La définition consiste à dire qu'il doit exister un hyperplan laissant d'un côté toutes les données positives et de l'autre, toutes les données négatives. Dès lors, on peut définir deux plans se trouvant de part et d'autre de l'hyperplan et parallèles à celui-ci, sur lesquels reposent les exemples les plus proches. La figure II.2 illustre cette situation.

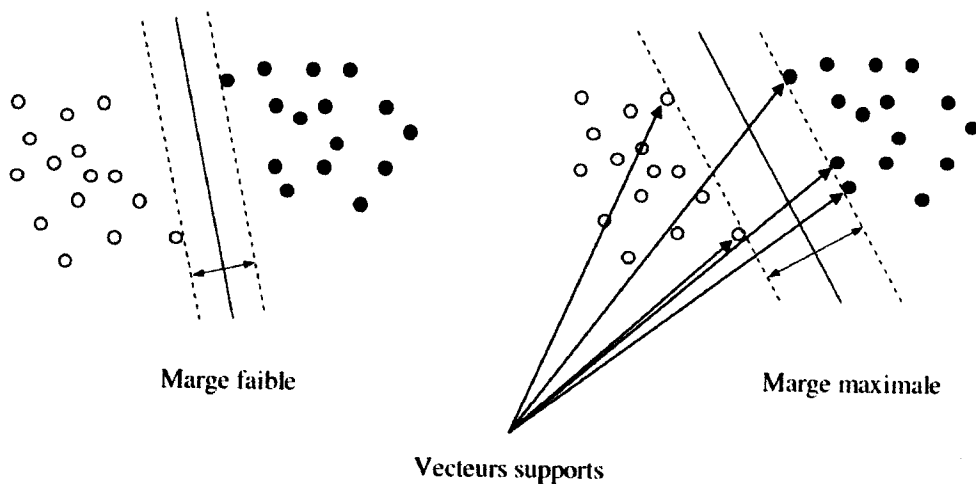


Figure II.2 : Illustration de la marge maximale et des vecteurs supports

Dans notre définition de l'hyperplan, il est possible que différentes équations correspondent au même plan géométrique :

$$a(\langle w, x \rangle + b) = 0 \quad (2.7)$$

$a$  est une constante quelconque.

Il est donc possible de redimensionner  $(w^*, b^*)$  de telle sorte que les deux plans parallèles aient respectivement pour équations :

$$(w^* x_i + b^*) = 1 \quad (2.8)$$

$$(w^* x_i + b^*) = -1 \quad (2.9)$$

Ces deux hyperplans sont appelés hyperplans canoniques. Ainsi la marge entre ces deux plans est égale à :

$$\gamma = \frac{2}{\|w^*\|} \quad (2.10)$$

#### II.2.4 Minimisation quadratique sous contraintes

Maintenant que nous avons défini les notions de marges et d'hyperplans canoniques, nous pouvons formuler un problème d'optimisation mathématique tel que sa solution nous fournisse l'hyperplan optimal qui permet de maximiser la marge [Boser92] :

$$\text{minimiser } \frac{1}{2} \|w\|^2 \quad (2.11)$$

$$\text{Tel que } y_i(\langle w, x_i \rangle + b) \geq 0$$

Il s'agit d'un problème quadratique convexe sous contraintes linéaires de forme primal dont la fonction objective est à minimiser. Cette fonction objective est le carré de l'inverse de la double marge. L'unique contrainte stipule que les exemples doivent être bien classés et qu'ils ne dépassent pas les hyperplans canoniques.

Dans cette formulation, les variables à fixer sont les composantes  $w_i$  et  $b$ . Le vecteur possède un nombre de composantes égal à la dimension de l'espace d'entrée. Généralement dans ce type de cas on résout la forme dual du problème. Nous devons former ce que l'on appelle le Lagrangien. Il s'agit de faire rentrer les contraintes dans la fonction objective et de pondérer chacune d'entre elles par une variable dual [Smola98]:

$$L(w, b, \alpha) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^n \alpha_i (y_i(\langle w, x_i \rangle + b) - 1) \quad (2.12)$$

Les variables duales  $\alpha_i$  intervenant dans le Lagrangien sont appelées multiplicateurs de Lagrange. Notons que  $L$  doit être minimisé par rapport aux variables primales  $w_i$  et  $b$  et maximisé par rapport aux variables duales  $\alpha_i$ .

Le point selle (minimal par rapport à une variable, maximal par rapport à l'autre) doit donc satisfaire les conditions nécessaires de stationnarité qui correspondent aux conditions Karush Kuhn et Tucker (KKT), nous trouvons:

$$\frac{\partial L(w,b,\alpha)}{\partial w} = 0 \quad (2.13)$$

$$\frac{\partial L(w,b,\alpha)}{\partial b} = 0 \quad (2.14)$$

Ce qui nous permet d'obtenir :

$$w = \sum_{i=1}^n \alpha_i y_i x_i \quad (2.15)$$

$$\sum_{i=1}^n \alpha_i y_i = 0 \quad (2.16)$$

Remarquons qu'avec cette formulation, on peut calculer  $w$  en fixant seulement  $n$  paramètres. L'idée va donc être de formuler un problème dual dans lequel  $w$  est remplacé par sa nouvelle formulation. De cette façon, le nombre de paramètres à fixer est relatif au nombre d'exemples de l'échantillon d'apprentissage et non plus à la dimension de l'espace d'entrée. Pour se faire, nous substituons (2.15) et (2.16) dans le Lagrangien, nous obtenons le problème dual équivalent suivant :

$$\text{maximiser}_{\alpha} W(\alpha) = \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{j=1}^n y_i y_j \alpha_i \alpha_j \langle x_i, x_j \rangle \quad (2.17)$$

$$\text{Tel que } \sum_{i=1}^n \alpha_i y_i = 0$$

$$\alpha_i \geq 0$$

Ce dernier problème peut être résolu en utilisant des méthodes standards de programmation quadratique [Smola98]. Une fois la solution optimale  $w^* = (\alpha_1^*, \alpha_2^*, \dots, \alpha_n^*)$  du problème (2.17) obtenue, le vecteur de poids de l'hyperplan à marge maximale recherché s'écrit :

$$w^* = \sum_{i=1}^n \alpha_i^* y_i x_i \quad (2.18)$$

Comme le paramètre  $b$  ne figure pas dans le problème dual, sa valeur optimale  $b^*$  peut-être dérivée à partir des contraintes primales, soit donc :

$$b^* = \frac{\max_{y_i=-1} (\langle w^*, x_i \rangle) + \min_{y_i=1} (\langle w^*, x_i \rangle)}{2} \quad (2.19)$$

Une fois les paramètres  $b^*$  et  $\alpha^*$  calculés, la règle de classification d'une nouvelle observation  $x$  basée sur l'hyperplan à marge maximale est donnée par :

$$h(x) = (\sum_{i=1}^n y_i \alpha_i^* \langle x_i, x \rangle + b^*) \quad (2.20)$$

Notons qu'un grand nombre de termes de cette somme est nul. En effet, seuls les  $\alpha_i^*$  correspondants aux exemples se trouvant sur les hyperplans canoniques (sur la

contrainte) sont non nuls. Ces exemples sont appelés Supports Vecteurs (SV). On peut les voir comme les représentants de leurs catégories car si l'échantillon d'apprentissage n'était constitué que des SV, l'hyperplan optimal que l'on trouverait serait identique.

### II.2.5 SVM non-linéaire

Les paragraphes précédents décrivent le principe des SVM dans le cas où les données sont linéairement séparables. Cependant, dans la plupart des problèmes réels, ce n'est pas toujours le cas et il est donc nécessaire de contourner ce problème (difficile de séparer n'importe quel jeu de données par un simple hyperplan). Si par exemple les données des deux classes se chevauchent sévèrement, aucun hyperplan séparateur ne sera satisfaisant.

Dans ce but et selon [Boser92], l'idée est de projeter les points d'apprentissage  $x_i$  dans un espace  $T$  de dimension  $q$ , plus élevée que  $n$  grâce à une fonction non-linéaire  $\phi$  qu'on appelle fonction noyau, choisie a priori et d'appliquer la même méthode d'optimisation de la marge dans l'espace  $T$ . L'espace  $T$  ainsi obtenu est appelé espace des caractéristiques ou aussi espace transformé.

Tout ce qu'il nous reste à faire c'est de résoudre le problème (2.17) dans l'espace  $T$ , en remplaçant  $\langle x_i, x_j \rangle$  par  $\langle \phi(x_i), \phi(x_j) \rangle$ . L'hyperplan séparateur obtenu dans l'espace  $T$  est appelé hyperplan optimal généralisé. Le produit scalaire  $\langle \phi(x_i), \phi(x_j) \rangle$  peut se calculer facilement à l'aide d'une fonction symétrique  $K$ , dite noyau, définie par :

$$K(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle \quad (2.21)$$

Le noyau  $K(x, y)$  est une fonction de deux variables, symétrique et positive. Dans ce cas, la frontière de décision devient :

$$h(x) = \text{sign}(\sum_{i=1}^n y_i \alpha_i^* K(x_i, x) + b^*) \quad (2.22)$$

Dans la pratique on choisit un noyau  $K$  qui satisfait les conditions de Mercer [Dodge99] afin de garantir la décomposition (2.22). Une famille de ces fonctions noyaux qui sont très appropriées aux besoins des SVM peut être définie, en voici les plus utilisés [Dodge99]:

- Noyau polynomial d'ordre  $p$ :

$$K(x_i, x_j) = \langle x_i, x_j + 1 \rangle^d \quad (2.23)$$

La dimension de l'espace transformé induit par un noyau polynomial est de l'ordre  $\frac{(p+d)!}{p!d!}$  où  $p$  est la dimension de l'espace de départ

- Noyau linéaire :

$$K(x_i, x_j) = x_i \cdot x_j \quad (2.24)$$

- Noyau gaussien de largeur de bande :

$$K(x_i, x_j) = \exp\left(-\frac{\|x_i - y_j\|^2}{2\sigma}\right) \quad (2.25)$$

Le paramètre  $\sigma$  permet de régler la largeur de la gaussienne. En prenant un  $\sigma$  grand, la similarité d'un exemple par rapport à ceux qui l'entourent sera assez élevée, alors qu'en prenant un  $\sigma$  tendant vers 0, l'exemple ne sera similaire à aucun autre.

- Le noyau RBF (Radial Basis Function)

$$K(x_i, x_j) = \exp(-g \|x_i - y_j\|^2) \quad (2.26)$$

### II.2.6 Relaxation des contraintes

Quand le domaine du problème d'optimisation est vide et il n'admet donc pas de solution, dans ce cas les données sont non linéairement séparables. Pour tenter de résoudre ce problème, on relâche les contraintes (2.12) dans le but d'autoriser quelques erreurs de classification. Cette généralisation de l'hyperplan à marge maximale a été proposée par Cortes et al. [Cortes95] en introduisant les variables d'écart à la marge  $(\varepsilon_i)_{1 \leq i \leq n}$ . L'hyperplan optimal est celui qui satisfait les conditions suivantes :

- ⇒ La distance entre les vecteurs bien classés et l'hyperplan optimal doit être maximal.
- ⇒ La distance entre les vecteurs mal classés et l'hyperplan optimal doit être minimal.

Le problème (2.11) devient alors :

$$\text{minimiser } \frac{1}{2} \|w\|^2 + c \sum_{i=1}^n \varepsilon_i \quad (2.26)$$

$$\text{Tel que } y_i(\langle w, x \rangle + b) \geq 1 - \varepsilon_i, i = 1, 2, \dots, n$$

$$\varepsilon_i \geq 0, i = 1, 2, \dots, n$$

Autrement dit, on cherche à maximiser la marge en s'autorisant pour chaque contrainte une erreur positive  $\varepsilon_i'$ , la plus petite possible. Le paramètre supplémentaire  $C$  est appelé paramètre de pénalisation du relâchement.  $C$  est une constante positive fixée à l'avance qui permet de contrôler l'importance de l'erreur que l'on s'autorise par rapport à la taille de la marge. Plus  $C$  est important, moins d'erreurs sont autorisées.

### II.2.7 SVM pour le cas multi-classes

La plupart des problèmes ne se contentent pas de deux classes de données. Il existe plusieurs méthodes pour passer à la classification multiclassées. Les deux méthodes les plus utilisées sont:

- *La méthode Un-contre-Un* celle que nous avons utilisée dans notre application le principe est que chaque classe est discriminée d'une autre.
- *La méthode Un-Contre-Tous* qui étend la notion de marge aux cas multiclassées. Cette formulation permet de poser un problème d'optimisation unique. Ce qui fait intervenir  $N$  fonctions de décision

### II.2.8 Avantages et inconvénients

#### Avantages.

- Les SVM possèdent des fondements mathématiques solides.
- Les exemples de test sont comparés juste avec les supports vecteur et non pas avec tout les exemples d'apprentissage.
- La décision est rapide puisque la classification d'un nouvel exemple consiste à voir le signe de la fonction de décision  $f(x)$ .

#### Inconvénients

- C'est une classifications binaire, d'où la nécessité d'utiliser les approches un-contre-un ou un-contre-tous.
- Une grande quantité d'exemples en entrées implique un calcul matriciel important.
- Un temps de calcul élevé lors d'une régularisation des paramètres de la fonction noyau.



### **II.3 Réseaux de Neurones**

Le rêve de créer une machine dotée d'une forme d'intelligence est présent depuis longtemps dans l'imagination humaine. Alors comment l'homme fait-il pour raisonner, calculer, parler, apprendre, ... ? C'est ces questions là qui mènent les chercheurs à essayer de comprendre le fonctionnement du cerveau humain et essayer de s'y inspirer pour pouvoir trouver de nouvelles techniques de résolutions de problèmes dans le monde informatique.

L'intelligence artificielle est apparue et ne cesse de progresser, il existe de nombreux programmes capables de diriger un robot, prendre des décisions, ... etc. Néanmoins ils ne sont pas capables de rivaliser avec un cerveau humain. Outre la capacité de calcul incroyable des ordinateurs, mais ces derniers n'ont pas la faculté d'apprendre. Ils ne progressent pas si personne ne les modifie. Voilà ce à quoi les chercheurs ont essayé de remédier.

Avec l'avancée dans le domaine de la neurobiologie concernant le fonctionnement du cerveau et des neurones, des mathématiciens ont essayé de modéliser le fonctionnement du cerveau en intégrant ces connaissances en biologie dans des programmes informatiques pour leur donner la possibilité d'apprendre : c'est la naissance des réseaux de neurones [Atteia99].

#### **II.3.2 Neurone biologique**

Le système nerveux compte plus de 1000 milliards de neurones interconnectés. Les neurones ne sont pas tous identiques, ni dans leurs formes ni dans leurs caractéristiques. En effet les neurones n'ont pas tous un comportement similaire en fonction de leur position dans le cerveau [Atteia99]. La figure II.3 montre le schéma d'un neurone biologique.

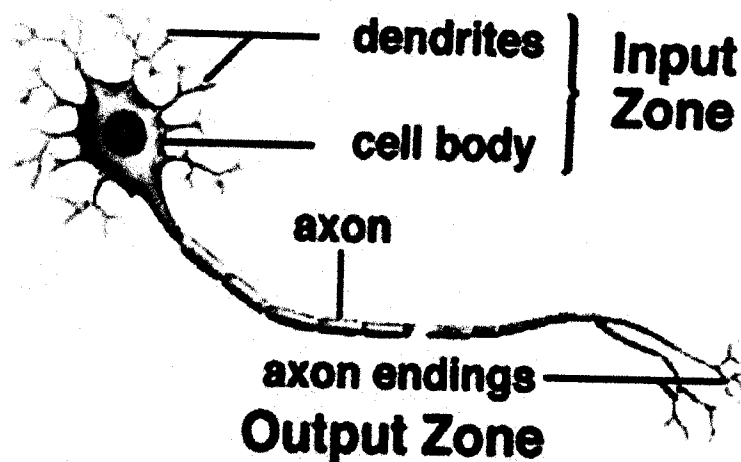


Figure II.3 : neurone biologique

### II.3.3 Principe de fonctionnement

L'information ne se transmet que dans un seul sens : des dendrites vers l'axone. Le neurone va donc recevoir des informations, venant d'autres neurones, grâce à ses dendrites. Il va ensuite y avoir une sommation, au niveau du corps cellulaire, de toutes ces informations et via un signal électrique. Le résultat de l'analyse va transiter le long de l'axone jusqu'aux terminaisons synaptiques. Il va y avoir libération des neurotransmetteurs (médiateurs chimiques) dans la fente synaptique. Le signal électrique ne pouvant pas passer la synapse, les neurotransmetteurs permettent donc le passage des informations, d'un neurone à un autre. Les neurotransmetteurs excitent ou inhibent le neurone suivant et peuvent ainsi générer ou interdire la propagation d'un nouvel influx nerveux. Les synapses possèdent une sorte de «mémoire» qui leur permet d'ajuster leur fonctionnement. En fonction de leur «histoire», c'est-à-dire de leur activation répétée ou non entre deux neurones, les connexions synaptiques vont se modifier : mécanisme d'apprentissage [Atteia99].

#### II.3.3.1 Neurone formel

Les réseaux de neurone formels ou artificiels sont des réseaux dont l'architecture est inspirée de celle des réseaux de neurones biologiques (naturels), ils sont généralement optimisés par des méthodes d'apprentissage de type statistique. Leur modélisation revient à décrire le modèle du neurone (unité de base) et le modèle des connexions entre les neurones. Le premier neurone formel

est apparu en 1943, introduit par MacCulloch et Pitts (unité à seuil). La figure (II.4) ci-dessous montre un schéma d'un neurone formel.

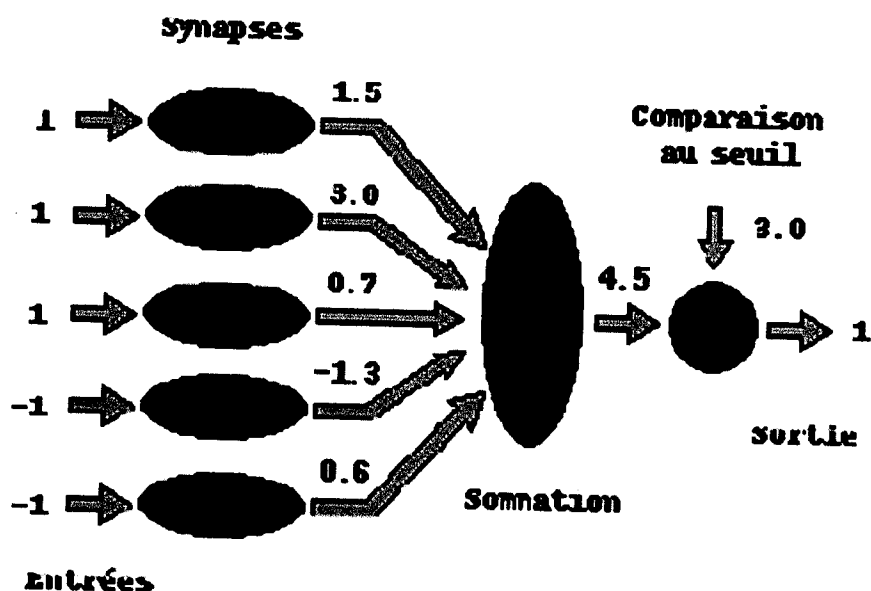


Figure II.4 : neurone formel (artificiel)

Le neurone formel est donc une modélisation mathématique qui reprend les principes du fonctionnement du neurone biologique, en particulier la sommation des entrées. Sachant qu'au niveau biologique, les synapses n'ont pas toutes la même valeur (les connexions entre les neurones étant plus ou moins fortes), les auteurs ont donc créé un algorithme qui pondère la somme de ses entrées par des poids synaptiques (sommation des multiplications des valeurs d'entrées par les coefficients de pondération). De plus sur le schéma, les 1 et les -1 en entrée sont pour différencier entre une synapse excitatrice ou inhibitrice, la sortie est obtenue après la comparaison de la sommation pondérée des entrées avec un seuil (si le résultat est supérieur au seuil, alors la valeur renvoyée est 1, sinon 0), d'où une fonction d'activation se présente (ici c'est une fonction à seuil).

Le choix d'une fonction d'activation se révèle être un élément constitutif important dans des réseaux de neurones. A titre illustratif voici quelques fonctions couramment utilisées comme fonctions d'activation : le sigmoïde standard (encore appelé fonction logistique), la tangente hyperbolique, la fonction gaussienne, une fonction à seuil.

### II.3.3.2 Calcul des poids synaptiques

La rétro-propagation est une méthode de calcul des poids (aussi appelés poids synaptiques du nom des synapses, terme désignant la connexion biologique entre deux neurones) pour un réseau à apprentissage supervisé qui consiste à minimiser l'erreur quadratique de la sortie (somme des carrés de l'erreur de chaque composante entre la sortie réelle et la sortie désirée).

D'autres méthodes de modification des poids sont plus locales, chaque neurone modifie ses poids en fonction de l'activation ou non des neurones proches.

### II.3.4 Quelques réseaux célèbres

Il y a de très nombreuses sortes de réseaux de neurones actuellement. De nouveaux réseaux (ou du moins des variations de réseaux plus anciens) sont inventés chaque semaine. On en présente ici de très classiques [Pierre07].

#### II.3.4.1 Le perceptron

C'est l'un des premiers réseaux de neurones, conçu en 1958 par Rosenblatt. Il est linéaire et monocouche. Il est inspiré du système visuel. La première couche (d'entrée) représente la rétine. Les neurones de la couche suivante (unique, d'où le qualificatif du monocouche) sont les cellules d'association, et la couche finale les cellules de décision. Les sorties des neurones ne peuvent prendre que deux états (-1 et 1 ou 0 et 1). Seuls les poids des liaisons entre la couche d'association et la couche finale peuvent être modifiés. La règle de modification des poids utilisée est la règle de *Widrow-Hoff*: si la sortie du réseau (donc celle d'une cellule de décision) est égale à la sortie désirée, le poids de la connexion entre ce neurone et le neurone d'association qui lui est connecté n'est pas modifié.

Dans le cas contraire le poids est modifié proportionnellement. À la différence entre la sortie obtenue et la sortie désirée :  $w \leq w + k(d - s)$  Où  $s$  est la sortie obtenue,  $d$  la sortie désirée et  $K$  une constante positive.

#### II.3.4.2 Les perceptrons multicouches (PMC)

Ils sont une amélioration du perceptron comprenant une ou plusieurs couches intermédiaires dites couches cachées, dans le sens où elles n'ont qu'une utilité intrinsèque pour le réseau de neurones et pas de contact direct avec l'extérieur. Chaque neurone n'est relié qu'aux neurones des couches précédentes et suivantes, mais à tous les neurones de ces couches (voir figure II.5).

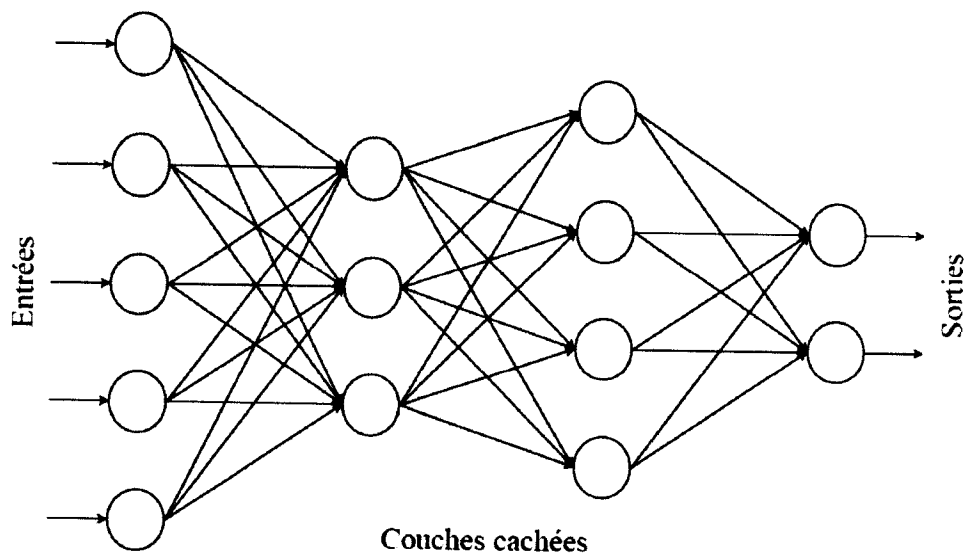


Figure II.5 : schéma d'un perceptron multicouche

Les PMC utilisent, pour modifier leurs poids, un algorithme d'apprentissage, il existe une centaine mais le plus populaire est la rétro propagation du gradient, qui est une généralisation de la règle de *Widrow-Hoff*. Il s'agit toujours de minimiser l'erreur quadratique, on propage la modification des poids de la couche de sortie jusqu'à la couche d'entrée, donc cet algorithme passe par deux phases :

- Les entrées sont propagées de couche en couche jusqu'à la couche de sortie.
- Si la sortie du PMC est différente de la sortie désirée alors l'erreur est propagée de la couche de sortie vers la couche d'entrée en modifiant les poids durant cette propagation.

#### II.3.4.3 Les réseaux de RBF (Radial Basis Fonction)

Le réseau RBF est un réseau de neurones supervisé. Il s'agit d'une spécialisation d'un PMC. Un RBF est constitué uniquement de 3 couches (figure II.6)

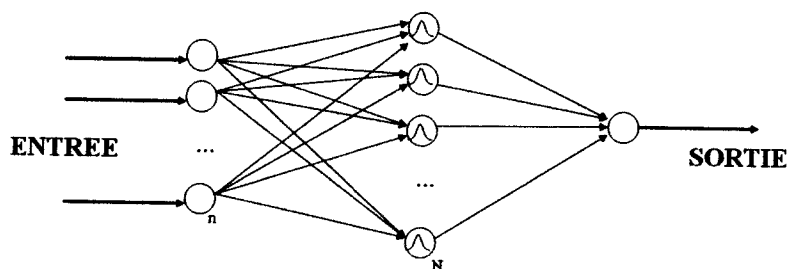


Figure II.6 : schéma d'un RBF

- La couche d'entrée : elle retransmet les entrées sans distorsion.
- La couche RBF : couche cachée qui contient les neurones RBF.
- La couche de sortie : simple couche qui contient une fonction linéaire.

Chaque neurone RBF contient une gaussienne qui est centrée sur un point de l'espace d'entrée. Pour une entrée donnée, la sortie du neurone RBF est la hauteur de la gaussienne en ce point. La fonction gaussienne permet aux neurones de ne répondre qu'à une petite région de l'espace d'entrée, région sur laquelle la gaussienne est centrée. Donc il y a quatre paramètres principaux à régler dans un réseau RBF.

- Le nombre de neurones RBF (nombre de neurones dans l'unique couche cachée).
  - La position des centres des gaussiennes de chacun des neurones.
  - La largeur de ces gaussiennes.
  - Le poids des connexions entre les neurones RBF et le(s) neurone(s) de sortie.
- Toute modification d'un de ces paramètres entraîne directement un changement du comportement du réseau.

### II.3.5 Avantages et inconvénients

#### Avantages

- Un classifieur très précis (s'il est bien paramétré).
- Apprentissage automatique des poids.
- Possibilité de faire le parallélisme (les éléments de chaque couche peuvent fonctionner en parallèle).
- Résistance aux pannes (si un neurone ne fonctionne plus, le réseau ne se perturbe pas).

#### Inconvénients

- Détermination de l'architecture du réseau est complexe.
- Paramètres difficiles à interpréter (boîte noire).
- Difficulté de paramétrage surtout pour le nombre de neurone dans la couche cachée.

## II.4 K plus proches voisins

K plus proches voisins est une méthode de raisonnement à partir de cas. Elle part de l'idée de prendre des décisions en recherchant un ou des cas similaires déjà résolus en mémoire. Contrairement aux autres méthodes de classification (Arbres de Décision, Réseaux de Neurones, Algorithmes Génétiques) [Grappa3].

C'est l'une des méthodes les plus simples de l'apprentissage automatique supervisé (Apprentissage ultra simple; Pas d'apprentissage simplement stockage de données d'apprentissage). C'est une méthode qui est basé sur un algorithme de la famille des algorithmes dits « paresseux » à l'inverse de beaucoup d'autres méthodes d'apprentissage automatique (les réseaux de neurones artificiels, les méthodes à noyaux, etc.).

### II.4.1 Principe du classifieur KPPV

Etant donnée une base d'exemples, pour prédire la classe d'un nouvel exemple, le classifieur KPPV cherche les K plus proches voisins de ce nouvel exemple et prédit la réponse la plus fréquente de ces K plus proches voisins. La méthode utilise donc deux paramètres : le nombre K et la fonction de similarité pour comparer le nouveau cas aux cas déjà classés. Le principe est donné par :

1. choix d'un entier k (souvent  $k = \text{nombre d'attributs} + 1$ , ou  $k = \sqrt{n}$ ).
2. calcul des distances (ex: distance euclidienne, c'est la distance la plus populaire).
3. retenir les k observations pour lesquelles ces distances sont les plus petites (les k plus proches voisins d'un cas de référence; quand on parle de voisin cela implique la notion de distance ou de dissimilarité).
4. compter les nombres de fois où ces k observations apparaissent dans chacune des classes (déterminer les classes correspondantes).
5. Choisir la classe la plus représentée.

Ces étapes sont résumées comme suit :

Pour une forme inconnue x à classer, nous allons examiner la distance de x à tous les échantillons (qui définissent toutes les classes), puis nous sélectionnons les K plus proches échantillons et nous affectons x à la classe majoritaire parmi ces K échantillons (figure 3.1)

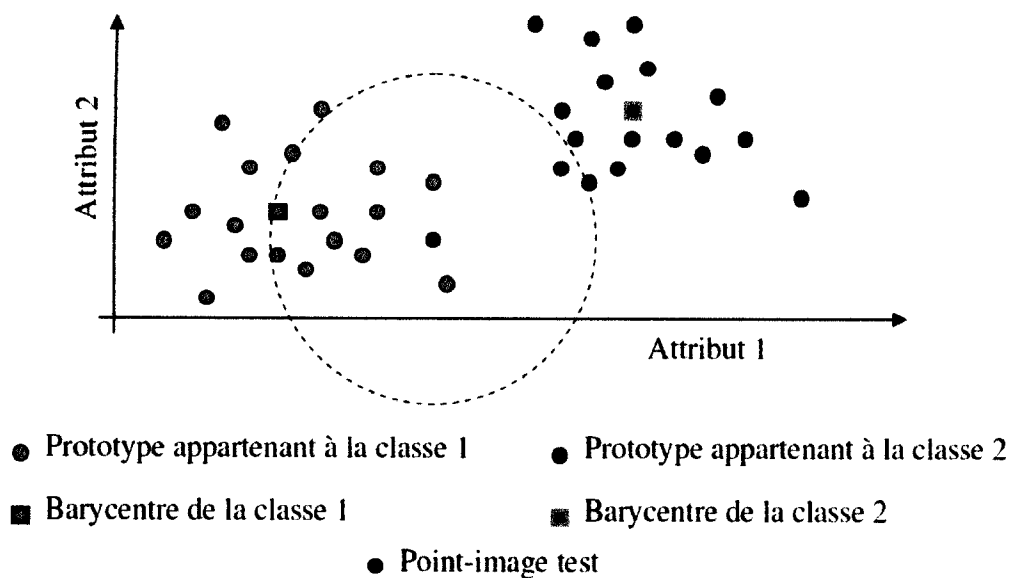


Figure II.7. Méthode de Kplus proches voisins

#### II.4.2 Choix du nombre K

Le choix du paramètre K dans la règle des KPPV a une influence directe sur le style du classifieur ainsi déterminé. Une faible valeur de K va donner un classifieur de bonne résolution (définir des frontières compliquées entre classes) mais très sensible au bruit sur les échantillons et sur le vecteur à classer. Une valeur grande de K aura un comportement inverse, lissant les frontières mais peu sensible au bruit [Milgram93].

#### II.4.3 Définition de la distance

Le choix de la distance est primordial au bon fonctionnement de la méthode.

Quoique les distances les plus simples permettent d'obtenir des résultats satisfaisants [Grappa3].

Une distance doit avoir quatre propriétés pour tous les vecteurs a, b et c [Duda92] :

- $D(a, b) \geq 0$  (non-négativité)
- $D(a, b) = 0$  si et seulement si  $a = b$  (réflexivité)
- $D(a, b) = D(b, a)$  (symétrie)
- $D(a, b) + D(b, c) \geq D(a, c)$ . (triangle inégalité)

On peut cependant noter qu'un point « a » peut avoir un plus proche voisin « b » tandis que « b » possède de nombreux voisins plus proches que « a ».

Le choix de la distance se fait en fonction des connaissances préalables du problème. Il est possible de choisir la distance en faisant varier cette distance et,



pour chacun des choix, estimer l'erreur réelle. On choisit alors la distance donnant la meilleure erreur réelle estimée. Plusieurs types pour les distances :

- distance Euclidienne.
- distance "city block" ou 'Manhattan' (somme des valeurs absolues)
- distance de Tchebycheff.
- distance de Mahalanobis
- .....

Dans notre cas nous avons utilisé la distance Euclidienne (la distance la plus populaire), sa formule en « d » dimensions est la suivante :

$$D(a, b) = \left( \sum_{k=1}^d (a_k - b_k)^2 \right)^{\frac{1}{2}} \quad (2.30)$$

#### II.4.4 Sélection de la classe

La méthode la plus simple est de rechercher le cas le plus proche et de prendre la même décision. C'est la méthode 1-PPV (1-NN) du plus proche voisin. Si cette méthode peut fournir de bons résultats sur des problèmes simples pour lesquels les points sont bien répartis en groupes denses de même classe, en règle générale, il faut considérer un nombre de voisins plus important pour obtenir de bons résultats [Atteia99].

Une première façon de combiner les k classes des k voisins les plus proches est le vote majoritaire. Elle consiste simplement à prendre la classe majoritaire.

Une seconde façon est le vote majoritaire pondéré. Chaque vote, c'est-à-dire chaque classe d'un des k voisins sélectionnés, est pondéré. Le poids est inversement proportionnel à la distance entre le cas à classer et les autres k plus proches voisins classes (Une telle pondération s'appelle un noyau).

Dans les deux cas précédents, il est possible de définir une confiance dans la classe attribuée égale au rapport entre les votes gagnants et le total des votes.

#### II.4.5 Avantages et inconvénients

##### Avantages

- très simple à mettre en œuvre et d'utiliser directement l'ensemble d'apprentissage
- La qualité de la discrimination par cette méthode dépend du choix du nombre  $k$  de voisins Considérés

### **Inconvénients**

- Il est cependant souvent nécessaire de faire varier ce nombre  $k$  pour obtenir les meilleurs résultats possibles
- nécessite un espace mémoire très important pour stocker les données et pour faire les différents calculs dans la phase de classification
- A chaque nouvel individu à classer, il est nécessaire de parcourir tout l'ensemble d'apprentissage.

### **II.5 Conclusion**

Plusieurs classifieurs peuvent donner des résultats différents sur l'assignation d'un pixel à une classe donnée. Ceci est dû au modèle de décision qui diffère selon le classifieur considéré [Chindaro05].

L'utilisation simultanée de plusieurs méthodes et de plusieurs paramétrages permet le plus souvent de cumuler les avantages de chaque classifieur et d'obtenir ainsi un résultat de classification plus performant.

La décision finale peut être obtenue de différentes manières de combinaison et c'est ce que nous allons voir dans le chapitre suivant.

# **Chapitre III :**

## **Fusion d'information**

### **III.1 Introduction**

Depuis quelques années des méthodes de fusion ont été adaptées et développées pour diverses applications telles que le traitement du signal, le traitement d'image, l'indexation des documents, et bien d'autres.

Plusieurs sens sont donnés à la fusion d'informations, nous reprenons ici la définition proposée par [Bloch03] : La fusion d'informations consiste à combiner des informations issues de plusieurs sources afin d'aider à la prise de décision.

Nous ne cherchons pas ici à réduire les redondances contenues dans les informations issues de plusieurs sources, mais au contraire à en tenir compte afin d'améliorer la prise de décision. De même nous cherchons à modéliser au mieux les différentes imperfections des données (imprécisions, incertitudes, conflit, ambiguïté, incomplétude, fiabilité des sources, ...) non pas pour les supprimer, mais encore pour l'aide à la décision.

Dans ce travail, nous considérons une application dans le cadre de la classification. Plusieurs classifieurs peuvent fournir une information sur la classe de l'objet observé. Ainsi, nous retenons des approches de fusion pour résoudre un tel problème.

### **III.2 Typologies et niveaux de la fusion**

Un système de fusion est généralement composé de sources d'information, de moyens d'acquisition d'information, de moyens de communication et de capacités à traiter l'information. Il peut être par conséquent très complexe. Il est fréquent et pratique, lors de l'étude ou de la présentation d'un système, de séparer les aspects topologiques et les aspects traitement d'informations, même s'il existe des interconnexions. La topologie a une influence importante sur le choix de l'architecture du système de fusion, sur les choix d'outils, des méthodes de traitement et de communication [Arif05].

On peut trouver dans la littérature plusieurs manières de classer les différentes étapes ou types de fusion. Cette différence provient principalement du niveau où l'opération de fusion est accomplie, de l'objectif de cette opération, du type de sources (ou capteurs) et de l'application considérée.

On peut par exemple reprendre la classification générale proposée par [Dasarathy97] qui considère trois principaux niveaux de fusion :

- **fusion de mesures ou des pixels en traitement d'images**, on dit aussi bas niveau ; il concerne la fusion d'informations directement issues des capteurs.

- **fusion d'attributs**, on dit aussi niveau intermédiaire ; il concerne la combinaison d'informations extraites après diverses phases de traitement et d'analyse des mesures.
- **fusion de décisions**, on dit aussi haut niveau, celui-ci concerne la combinaison des décisions obtenues à partir de chaque source. La décision finale est prise en fonction des décisions de tous les capteurs.

Dans trois niveaux de fusion sont décrits de la manière suivante (dans le domaine de la fusion de classifieurs):

- (i) **Fusion au niveau de la représentation**, un classifieur  $e$  attribue à chaque classe des scores traduisant la similarité calculée entre la forme entrante  $x$  et les autres formes de la classe ou en donnant un certain degré de ressemblance entre telle ou telle classe.
- (ii) **Fusion au niveau des rangs**, un classifieur  $e$  donne les rangs de toutes les classes, le premier rang étant réservé à son choix préféré.
- (iii) **Fusion au niveau de la décision**, un classifieur  $e$  indique sa décision d'une classe ou parfois un sous-ensemble de classes.

L'avantage de la fusion bas niveau est qu'elle permet de combiner des données avant qu'elles ne soient déformées par les procédures de traitement et d'analyse. Par contre, la fusion aux niveaux intermédiaire et haut permet de se rapprocher davantage du raisonnement de l'expert.

D'après [Dasigi01] le besoin de "fusion" d'informations se fait sentir dans des problèmes de reconnaissance de formes où des décisions doivent être prises sur la base de différents ensembles de caractéristiques multiples dérivées de l'entrée. Quand l'information des ensembles de caractéristiques est assimilée et intégrée dans une décision finale, la technique est désignée généralement sous le nom de fusion d'informations. En revanche, si différentes décisions sont prises à partir des différents ensembles de caractéristiques, si l'on résout les éventuels conflits et qu'on intègre ces décisions dans une décision globale, la technique s'appelle fusion de décisions. La fusion de décisions est un problème très intéressant et important dans la recherche, elle constitue un domaine à part entière. Dans la communauté de la recherche documentaire, la fusion d'informations s'appelle parfois la combinaison de doutes, et la fusion de décisions s'appelle parfois la combinaison de données [Wald02].

### III.3 Stratégies de fusion

Les stratégies ou les architectures de fusion décrivent l'ensemble des sources, la manière dont elles sont assemblées et les techniques mathématiques ou statistiques pour le traitement.

La multiplication des travaux sur la fusion dans les domaines différents (imagerie, intelligence artificielle et reconnaissance de formes, etc.) a entraîné la mise au point de nombreux schémas traitant les données de manières différentes. Les stratégies de fusion proposées peuvent être regroupées en trois approches principales : séquentielle [Gader91], [Rahman03], parallèle [Sabourin94], [Huang95], [Gader96], [Kim00], [Zouari04], et hybride [Gosselin97], [Kim00], [Bellili02]. D'autres stratégies avec bouclage ou avec interaction sont aussi proposées. Malgré la diversité des schémas de fusion, la détermination de la meilleure organisation reste un problème ouvert, en effet, la variété des applications de fusion est telle qu'il n'est pas facile de conclure à une stratégie générique et applicable à tous les cas. D'après [Zouari04], l'intérêt porté par les chercheurs majoritairement à la combinaison parallèle de sources (classifieurs) est fondé sur plusieurs raisons :

- le concepteur peut réutiliser les développements de classifieurs effectués antérieurement, chacun pouvant avoir été développé dans un contexte différent et utiliser des représentations différentes qui peuvent être utilisées pour le même problème. Des exemples sont donnés par l'identification de personnes par leur voix, leur visage, ainsi que leur signature.
- Dans la combinaison, il est possible d'utiliser un grand nombre de caractéristiques mais en les distribuant sur des classifieurs différents.
- Deux classifieurs différents peuvent présenter des performances globales équivalentes mais se différencier par les régions où ils sont les plus performants.
- Un classifieur est souvent sensible aux choix initiaux de ses paramètres ( $k$  et la distance pour un  $k$ -ppv, nombre de couches et nombre de neurones par couche pour un perceptron multicouche, ...). Plutôt que de chercher la meilleure configuration de paramètres, la combinaison de l'ensemble peut tenir compte des avantages de ces classifieurs appris indépendamment.
- Nous pouvons disposer de plusieurs bases d'apprentissage, chacune est collectée de manière différente ou construite dans des conditions différentes,

l'apprentissage d'un même classifieur sur ces bases peut produire des résultats différents.

Voici le schéma de fusion de données fournies par des sources (classifieurs) différentes avec une approche de combinaison parallèle :

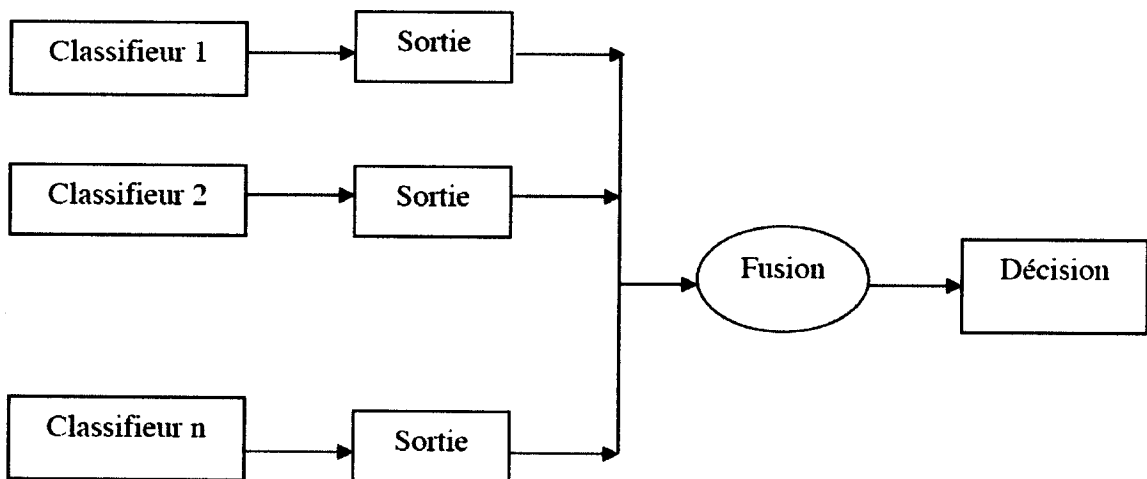


Figure III.1 : Une architecture de fusion de classifieurs en parallèle.

#### III.4 Etapes de l'opération de fusion

Pour un niveau hiérarchique donné, l'opération de fusion s'effectue en plusieurs étapes [Oxenham96], [Hall97] et [Wald02].

- i) **Alignement** : l'alignement ou conditionnement ou encore parfois, harmonisation, consiste à définir un espace commun, dans lequel les informations vont être projetées afin d'y être comparables. Cela veut dire que les observations ou les données sont ramenées dans un même référentiel.
- ii) **Corrélation** : cette étape concerne la détermination des relations entre les différentes données. Dans notre cas, nous ne sommes pas concernés par cette étape car nos classifieurs sont établis indépendamment et ils sont supposés non corrélés.
- iii) **Association ou mise en correspondance** : l'association est l'union des différentes représentations issues des informations multi-sources. Chaque mesure se trouve associée à l'entité correspondante (le résultat de l'étape de corrélation est évidemment utilisé). Cette étape permet aussi de rejeter les données aberrantes suivant un critère sur la matrice de covariance par exemple.

iv) Combinaison : seules les données obtenues après alignement et qui sont en accord avec l'étape d'association sont combinées pour obtenir une meilleure représentation de l'estimation correspondant à l'attribut avec lequel l'étape i) a été réalisée.

On peut noter par ailleurs que l'étape (ii) peut être imbriquée à l'intérieur de (iii). Par conséquent, trois étapes sont nécessaires pour toute opération de fusion : alignement, mise en correspondance (ou association) et combinaison [Oussalah98].

[Oussalah98] a distingué en plus les notions de fusion statique et de fusion dynamique en définissant :

- Fusion statique : quand le résultat de l'opération de fusion est obtenu indépendamment des états antérieurs. On exploite alors uniquement les données de l'instant courant.
- Fusion dynamique : quand le résultat tient compte des états antérieurs. Tout processus de fusion ayant une formulation itérative rentre dans ce contexte. Cette approche peut être située au même niveau que la fusion temporelle.

### III.5 Les méthodes de fusion

Nous présentons ici trois cadres théoriques de fusion d'information haute niveau, le principe du vote, la théorie des possibilités et la théorie de probabilité. Nous considérons le problème de la fusion de  $m$  sources  $S_j$  afin de déterminer une des  $n$  classes  $C_i$  possibles.

#### III.5.1 Principe du vote

Le principe du vote est la méthode de fusion d'informations la plus simple à mettre en œuvre. Plus qu'une approche de fusion, le principe du vote est une méthode de combinaison particulièrement adaptée aux décisions de type symbolique. Notons  $S_j(x) = i$  le fait que la source  $S_j$  attribue la classe  $C_i$  à l'observation  $x$ . Nous supposons ici que les classes  $C_i$  sont exclusives. A chaque source nous associons la fonction indicatrice :

$$M_i^j(x) = \begin{cases} 1 & \text{si } S_j(x) = i \\ 0 & \text{si } \text{non} \end{cases} \quad (3.1)$$

La combinaison des sources s'écrit par :

$$M_k^E(x) = \sum_{j=1}^k M_k^j(x) \quad (3.2)$$

Pour tout  $k$ , l'opérateur de combinaison est donc associatif et commutatif. La règle du vote majoritaire consiste à choisir la décision prise par le maximum de sources, c'est-à-



dire le maximum de  $M_k^E$ . Cependant cette règle simple n'admet pas toujours de solutions dans l'ensemble des classes  $= \{C_1, \dots, C_n\}$ . En effet, par exemple si le nombre de sources  $m$  est paire et que  $m/2$  sources décident  $C_{i_2}$  et  $m/2$  autres sources disent  $C_{i_2}$ , ou encore dans le cas où chaque source affecte à  $x$  une classe différente. Nous sommes donc obligé d'ajouter une classe  $C_{n+1}$  qui représente l'incertitude totale liée au conflit des sources sous l'hypothèse de l'exhaustivité des classes  $C_{n+1} = \{C_1, \dots, C_n\}$ . La décision finale de l'expert prise par cette règle s'écrit donc par :

$$E(x) = \begin{cases} k & \text{si } \max_k M_k^E(x) \\ n+1 & \text{sinon} \end{cases} \quad (3.3)$$

Cette règle est cependant peu satisfaisante dans les cas où deux sources donnent le maximum pour des classes différentes. La règle la plus employée est la règle du vote majoritaire absolu qui s'écrit :

$$E(x) = \begin{cases} k & \text{si } \max_k M_k^E(x) > \frac{m}{2} \\ n+1 & \text{sinon} \end{cases} \quad (3.4)$$

A partir de cette règle il a été démontré que la méthode du vote permet d'obtenir de meilleurs performances que toutes les sources prises séparément, sous des hypothèses d'indépendance statistique des sources et de même probabilité, et ceci est d'autant plus vrai que  $m$  est impaire.

Il est possible de généraliser le principe du vote majoritaire afin de supprimer le conflit. Au lieu de combiner les réponses des sources par une somme simple comme dans l'équation (2), l'idée est d'employer une somme pondérée [Krzyzak92] :

$$M_k^E(x) = \sum_{j=1}^k \alpha_{jk} M_k^j(x) \quad (3.5)$$

Où  $\sum_{j=1}^m \sum_{k=1}^n \alpha_{jk} = 1$ . Les poids  $\alpha_{jk}$  représentent la fiabilité d'une source pour une décision donnée, et l'estimation de ces poids peut se faire à partir des taux normalisés de réussite pour chaque classe et chaque classifieur. Notons qu'alors nous introduisons une connaissance *a priori* non nécessaire précédemment. Les différentes règles de décision possibles peuvent se résumer par la formule suivante :

$$E(x) = \begin{cases} k & \text{si } M_k^E(x) = \max_i M_i^E(x) \geq cm + b(x) \\ n+1 & \text{sinon} \end{cases} \quad (3.6)$$

Où  $c$  est une constante entre  $[0, 1]$  et  $b(x)$  est une fonction de  $M_k^E(x)$ .

### III.5.2 Théorie des probabilités

La théorie des probabilités est sans doute la théorie la plus classique et la plus ancienne pour traiter l'incertitude et pour fusionner les informations. Soit  $A$  un élément quelconque de  $\Omega$ . La probabilité que l'événement  $A$  soit vrai (ou pertinent) peut être quantifié par la quantité  $p(A)$ . On définit pour tout  $A$  de  $\Omega$  une fonction  $p$ , appelée distribution de probabilité par :

$$\begin{aligned} p : \Omega &\rightarrow [0,1] \\ A &\rightarrow p(A) \end{aligned} \quad (3.7)$$

Vérifiant les axiomes suivants :

$$\begin{aligned} p(\Omega) &= 1 \\ p(\emptyset) &= 0 \\ p(A \cup B) &= p(A) + p(B) \text{ si } B \subseteq \Omega \text{ et } A \cap B = \emptyset \end{aligned} \quad (3.8)$$

Le couple  $(\Omega, p)$  est appelé espace probabilisé. Ce dernier axiome permet de mettre en évidence que la mesure de probabilité est une mesure additive et donc la relation suivante est vérifiée :

$$\sum_{i=1}^q p(C_i) = 1 \quad (3.9)$$

L'un des principaux outils de la théorie des probabilités est sans aucun doute le théorème de Bayes qui permet la fusion d'informations entre sources. Cette probabilité s'écrit :

$$P(C_i/e(1) = C_1, \dots, e(k) = C_k) = P(C_i) \prod_{k=1}^K \frac{p(C_i/e(k) = C_k)}{P(C_{ji})} \quad (3.10)$$

$p(C_i/e(k) = C_k)$  et  $P(C_{ji})$  sont estimées à partir de la matrice de confusion obtenue pour le classifieur  $k$ .

La modélisation probabiliste n'apparaît pas toujours suffisante ni très bien adaptée pour représenter l'incertitude et modéliser les méconnaissances [45, 12]. De plus cette théorie

ne permet pas la gestion des conflits entre les sources puisque aucune autre information que la probabilité a posteriori n'est disponible.

### III.5.3 Théorie des possibilistes

La théorie des possibilités proposée par (Dubois et Prade [Dubois88]) permet de tenir compte de l'imprécision des données ainsi que de l'incertitude à partir de deux fonctions de possibilité et de nécessité. Ces deux fonctions sont obtenues à partir des distributions de possibilités définies sur  $D = \{C_1, \dots, C_n\}$  par :

$$\Pi: D \rightarrow [0,1], \sup_{x \in D} \pi(x) = 1 \quad (3.11)$$

Ces distributions donnent le degré d'appartenance au domaine  $D$ , qui n'est autre qu'un opérateur flou. Afin d'extraire l'imprécision et l'incertitude des données, deux fonctions spécifiques sont définies à partir de ces distributions. La fonction de possibilité est définie pour tout  $A \in 2^D$  par :

$$\Pi(A) = \sup_{x \in D} \pi(x) \quad (3.12)$$

La fonction de nécessité est donnée pour tout  $A \in 2^D$  par :

$$N(A) = 1 - \Pi(A^c) \quad (3.13)$$

Où  $A^c$  représente l'évènement contraire de  $A$ .

Un des avantages de la théorie des possibilités est le nombre d'opérateurs de combinaison disponibles. Il est ainsi possible de combiner l'information issue des distributions de possibilité, à partir d'opérateurs de type t-norme, t-conorme, moyenne, sommes symétriques, etc... Le choix du type de combinaison est un problème délicat *a priori* dans la théorie des possibilités, et doit être fait selon l'application et l'objectif recherché. Ce choix peut se faire selon le comportement général de l'opérateur (conjunctif, disjonctif, ou des compromis), selon les propriétés désirées, selon sa capacité à discriminer les classes, ou encore selon son comportement dans des situations de conflit. En pratique de nombreux opérateurs sont employés et testés dans les applications, tels que max (opérateur de type t-norme), min (opérateur de type t-conorme), ou la moyenne, la médiane et les intégrales floues (opérateurs de type moyenne).

La dernière étape de la fusion d'informations est l'étape de décision. Dans le cadre de la théorie des possibilités, elle est généralement faite selon la règle suivante : la classe  $C_k$  est décidée pour l'observation  $x$  si :

$$C_k = \operatorname{argmax}_{1 \leq i \leq n} \mu_i(x) \quad (3.14)$$

Où  $\mu_i(x)$  représente le coefficient d'appartenance de  $a$  à la classe  $C_i$ , qui sera ici donné par les sorties du classifieur.

Par construction des opérateurs de combinaison et de la règle de décision, la théorie des possibilités est davantage adaptée à la fusion d'informations de type numérique. Ainsi les coefficients d'appartenance peuvent être facilement obtenus dans le cadre de la classification par les sorties numériques des classifieurs. Nous emploierons donc ici cette théorie pour la fusion d'information haute niveau sur des données de type numérique.

### **III.6 Conclusion**

Dans ce chapitre, nous avons présenté les différents aspects que comporte la fusion de données. Nous avons vu qu'il existe plusieurs méthodes de fusion dont le choix dépend de la nature des sources et de l'information.

De nombreux travaux sur la fusion de données dans les applications réelles montrent que la fusion améliore nettement les performances du système de reconnaissance par rapport à chacune des sources prise isolément.

Après les généralités sur les méthodes et les techniques utilisées, nous passerons dans le chapitre suivant à l'expérimentation.

# **Chapitre IV:**

## **Application**

## **IV.1 Préliminaire**

Après les notions théoriques données dans les chapitres précédents, nous présentons dans ce dernier chapitre le côté pratique de notre application. Notre but est de réaliser un système qui segmente, d'une manière automatique, les images par la combinaison de différents classifieurs.

## **IV.2 Environnement de travail**

### **IV.2.1 Base d'images**

Notre base est construite à partir de 27 images cytologiques acquises au sein du service d'hémobiologie du C.H.U Tlemcen sur des lames avec la coloration de May Grunwald Giemsa(MGG).

L'environnement LEICA (caméra et microscope) permet d'obtenir des images couleur RGB de bonne qualité; au format BMP de dimension 1024x768.

Ce qui nous intéresse ici est de détecter 4 types de classes: globule blanc (noyau et cytoplasme), globule rouge et le fond (figure IV.1).

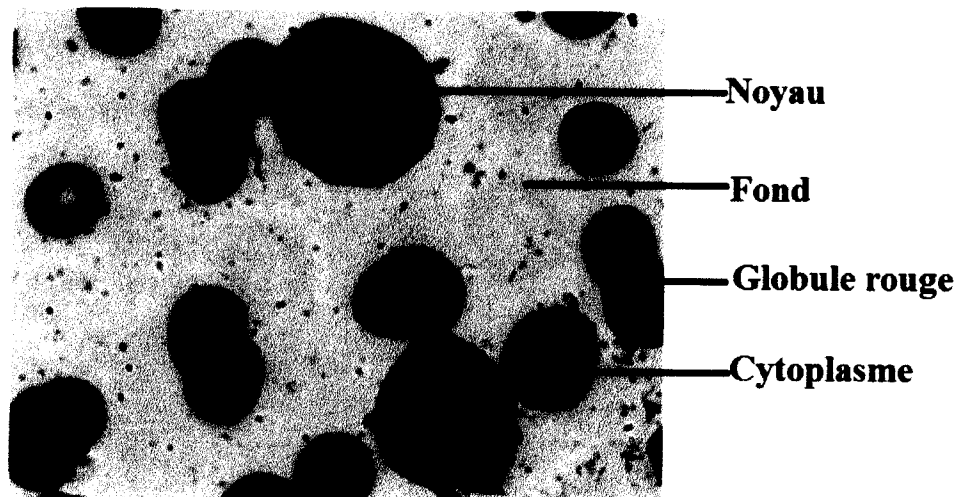


Figure IV.1:Exemple d'image cytologique

Les images ont été étiquetées en 3 phases (plus de détails dans [baghli11]). En plus des images étiquetées (figure IV.2), la base numérique est obtenue en convertissant les couleurs des classes en valeurs numériques ; soit 1 pour le vert (noyau), 2 pour le jaune (cytoplasme), 3 pour le rouge (globule rouge) et 4 pour le noir (fond).

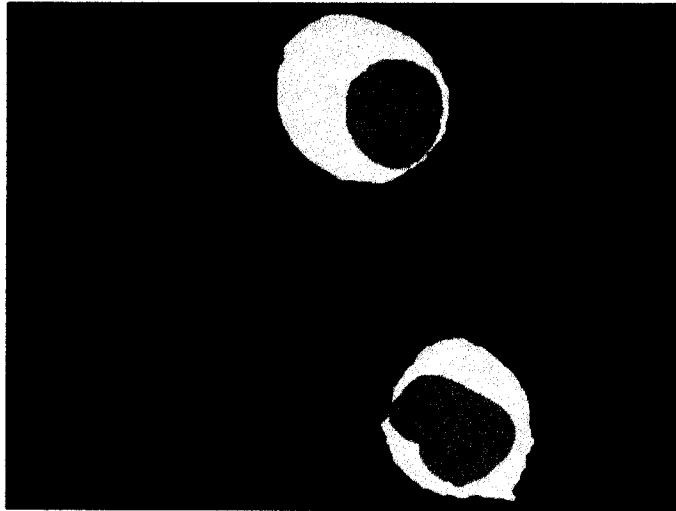


Figure IV.2:Image étiquetée

#### **IV.2.2 outil utilisé**

Pour nos expérimentations, nous avons optés pour le langage MATLAB qui est un langage de développement informatique particulièrement dédié aux applications scientifiques « traitement du signal, imagerie, etc. », d'où la justification du choix de ce langage qui nous permettra de traiter les images avec efficacité et rapidité. MATLAB est doté d'un environnement simple et convivial et contient de nombreuses boites à outils.

#### **IV.3 Phase d'apprentissage**

Initialement les images sont de type RGB (espaces de primaires) donc la première étape est de convertir l'image RGB aux autres espaces de couleur: HSL et HSV (espaces perceptuels), LUV (espace perceptuellement uniforme) et YUV (espace de télévision). Ensuite, la deuxième étape est de combiner les 5 espaces en une seule base de 1024x768 lignes (pixels) et 16 colonnes (3 colonne pour chaque espace et 1 colonne pour la classe).

### IV.3.1 Construction de la base d'apprentissage

Nous avons construits notre base d'apprentissage en prenant des régions noyau, des régions cytoplasme, des régions globule rouge et des régions fond (figureIV.3) de 9 images représentatives, cet ensemble contient 155010 pixels. Et sur cet ensemble nous avons construit les 3 classifieurs (KPPV, PMC et SVM).

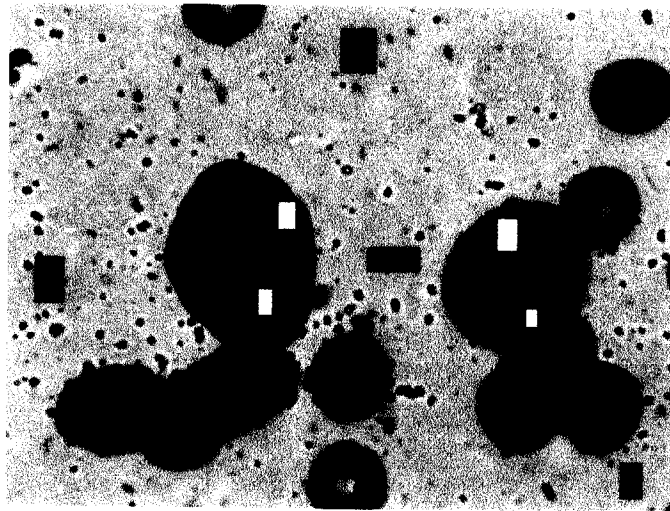


Figure IV.3: Définition des 4 régions

### IV.3.2 Apprentissage et test des classifieurs

Sur les 27 images testées, les taux de reconnaissance et la précision de chaque classe par rapport à chaque classifieur sont donnés dans les tableaux IV.1, IV.2 et IV.3

#### ✓ KPPV

Nous avons testé l'algorithme KPPV avec plusieurs valeur de K et à chaque fois le résultat différé, cependant il reste satisfaisant jusqu'à la valeur de K=9.

Précision				Taux			
Noyau	Cytoplasme	Rouge	Fond	Noyau	Cytoplasme	Rouge	Fond
0.9501	0.7826	0.7458	0.9790	0.9187	0.5976	0.9123	0.9733

TABLE IV.1 – Résultats obtenus pour le classifieur KPPV



✓ **PMC**

Après l'apprentissage du réseau de neurone, on a trouvé que l'erreur minimale est fixée à  $10^{-3}$  et le nombre maximal d'époques prévu à l'entraînement est égal à 80 époques.

Précision				Taux			
Noyau	Cytoplasme	Rouge	Fond	Noyau	Cytoplasme	Rouge	Fond
0.9435	0.8295	0.6429	0.9703	0.9341	0.7458	0.8491	0.9764

TABLE IV.2 – Résultats obtenus pour le classifieur PMC

✓ **SVM**

Nous avons construit le classifieur SVM à l'aide de l'implémentation multi-classes de [LIBSVM] avec une fonction noyau RBF (avec  $C=512$  et  $g=0.0001$ ) et une stratégie un contre un.

Précision				Taux			
Noyau	Cytoplasme	Rouge	Fond	Noyau	Cytoplasme	Rouge	Fond
0.9592	0.7633	0.8232	0.9755	0.8625	0.6378	0.9084	0.9765

TABLE IV.3 – Résultats obtenus pour le classifieur SVM

✓ **Performances des classifieurs**

Les 3 techniques utilisées ont donné des résultats différents puisqu'ils s'appuient sur des principes différents (voir chapitre 2). En considérant les précisions, nous avons constaté que le classifieur SVM se comporte mieux avec la classe Noyau et la classe Globule Rouge. Il en est de même pour la classe Fond avec le classifieur KPPV et la classe Cytoplasme avec le classifieur PMC. La figure IV.4 montre un des résultats des classifieurs.

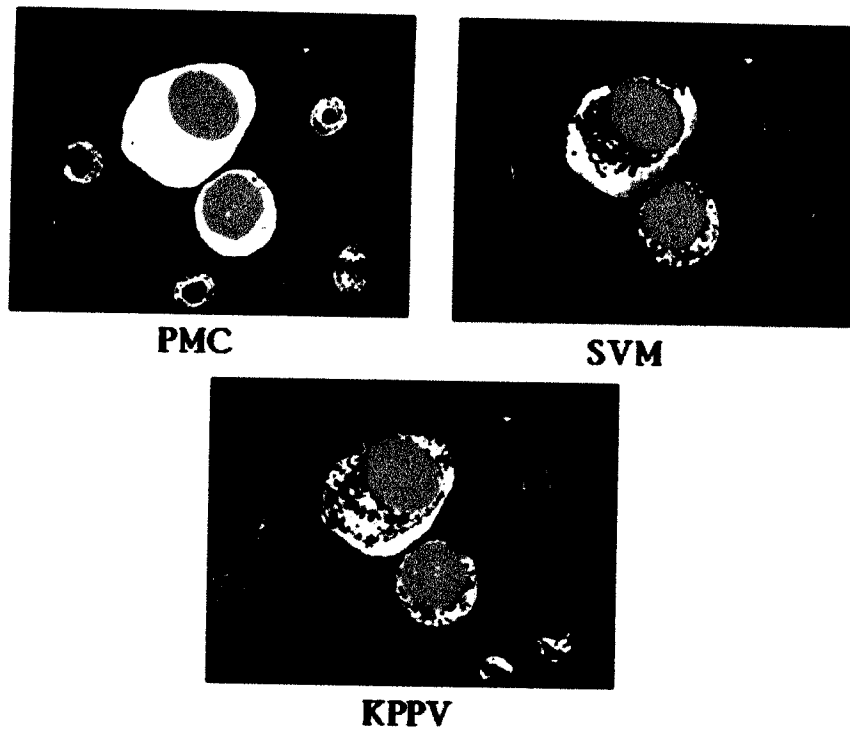


Figure IV.4: Résultats obtenus pour les trois classifieurs

#### IV.4 Phase de fusion

Tous les classifieurs utilisés ne donnent pas le même avis sur la classe à attribuer, nous obtenons des zones de non cohérence. L'étape de fusion résout ce problème en appliquant un formalisme de fusion (figure IV.5). Dans ce travail, nous avons expérimentés deux formalismes :

1. Vote majoritaire
2. Vote majoritaire pondéré

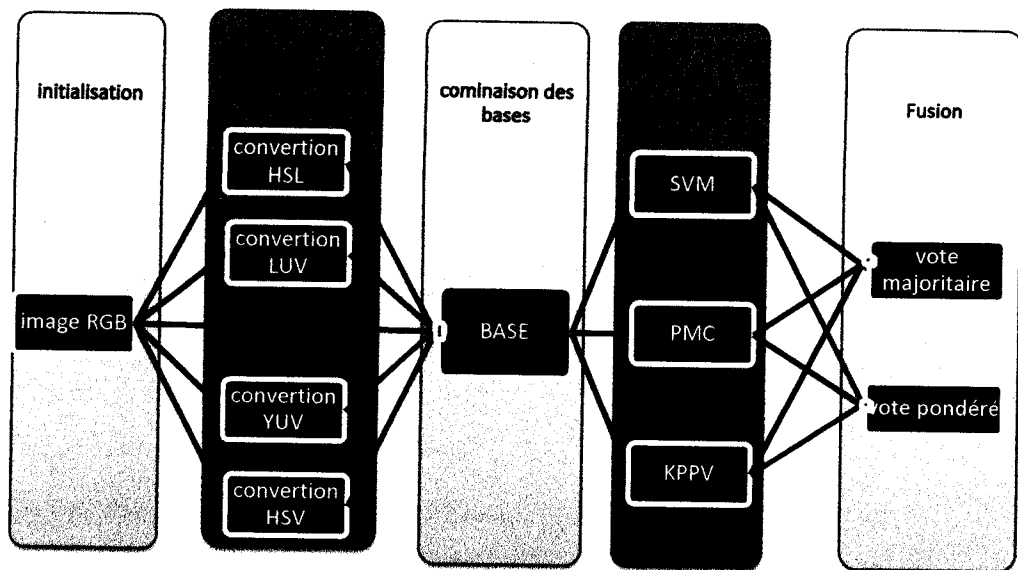


Figure IV.5: Stratégie de fusion

#### IV.4.1 Vote majoritaire

Dans cette première expérience, la classe attribuée au pixel incohérent est choisie par vote majoritaire. C'est-à-dire la classe la plus présente parmi celles présentant un conflit est élue à moins que les trois classifieur affecte chaque un une classe différente. Dans ce cas-là on choisit la décision du classifieur SVM vu qu'il est relativement meilleur par rapport au deux autres. Le tableau IV.6 contient les taux de reconnaissances ainsi que les précisions obtenus pour chaque classe.

Précision				Taux			
Noyau	Cytoplasme	Rouge	Fond	Noyau	Cytoplasme	Rouge	Fond
0.9504	0.8226	0.8033	0.9774	0.9172	0.6458	0.9150	0.9758

TABLE IV.6 : Résultats obtenus pour le vote majoritaire

#### IV.4.2 Vote majoritaire pondéré

Nous avons pondéré les sorties des classifieurs avec les précisions obtenues sur la base de test avec la condition de augmenter la précision si la classe X se comporte mieux avec le classifieur Y (Table IV.7).

	Noyau	Cytoplasme	Rouge	Fond
SVM	0.9592	0.6429	0.8232	0.9755
PMC	0.9435	0.8295	0.6429	0.9703
KPPV	0.9501	0.7826	0.7458	0.9790

TABLE IV.7: Précisions obtenues sur la base d'apprentissage

Le tableau IV.8 contient les taux de reconnaissances ainsi que les précisions obtenus pour chaque classe du vote majoritaire pondéré

Précision				Taux			
Noyau	Cytoplasme	Rouge	Fond	Noyau	Cytoplasme	Rouge	Fond
0.9562	0.8515	0.8099	0.9769	0.9234	0.6559	0.9163	0.9788

TABLE IV.8 – Résultats obtenus pour le vote majoritaire pondéré

Avec notre interface, développée sous Matlab, nous avons à choisir une image parmi les 27 images; pour affichez les résultats visuels obtenus par les trois classifieur (PMC, SVM et KPPV), le résultat du vote majoritaire et le résultat du vote majoritaire pondéré, en plus de l'image originale et l'image étiquetée (figure IV.6).

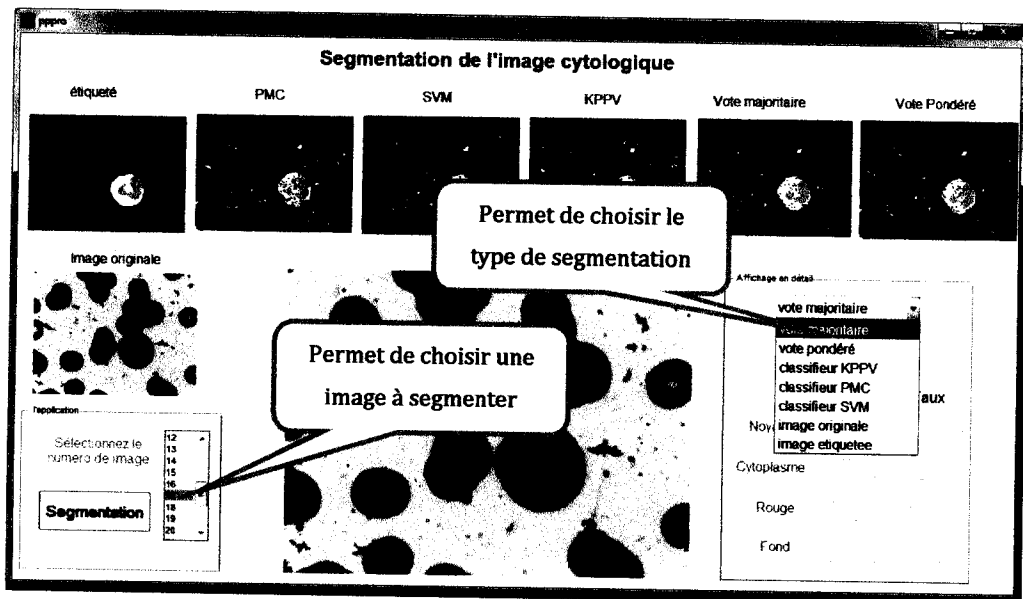


Figure IV.6: Interface du choix de l'image et de la segmentation

Et une zone pour agrandir le traitement en cours et afficher le taux de reconnaissances et la précision de chaque classe, comme c'est montré sur la figure IV.7

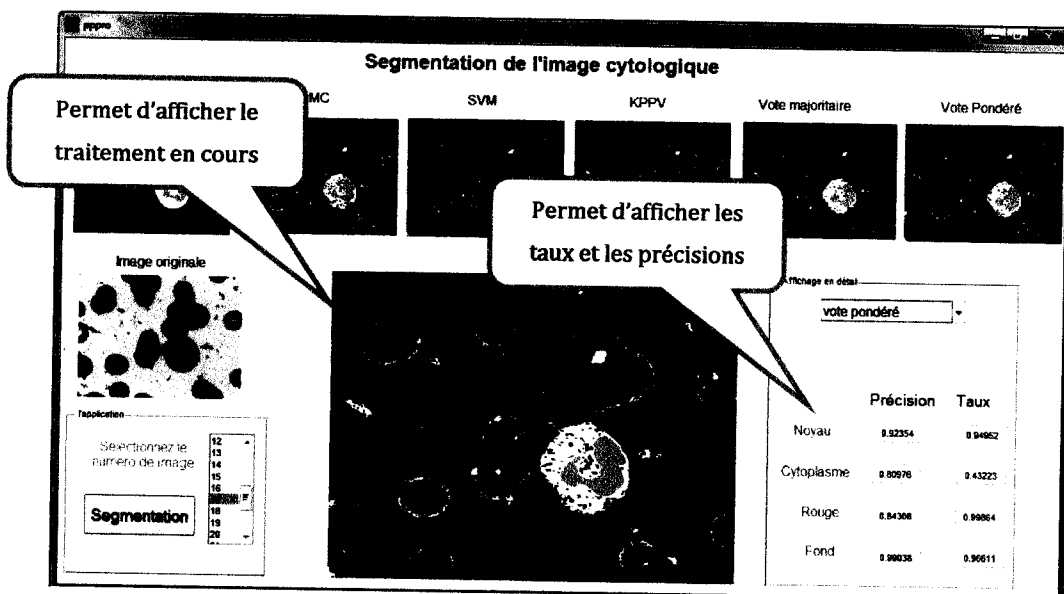


Figure IV.7: Affichage des détails du résultat

#### **IV.5 Discussion des résultats**

Sur la totalité des images testées et sur toutes les segmentations faites, nous avons constaté que :

1. les résultats obtenus pour toutes les classes sont assez satisfaisants en termes de taux de reconnaissances et de précisions. Le plus faible taux est celui de la classe "Cytoplasme" avec un pourcentage de 67.14% pour le vote majoritaire pondéré. Cela est dû principalement à l'écart enregistré sur les configurations colorimétriques de cette classe.
2. Chaque classifieur se comporte mieux avec une (deux) classe(s). Le classifieur SVM avec les classes "Noyau" et "Globules rouge". Le classifieur KPPV avec la classe "Fond" et le classifieur PMC avec la classe "Cytoplasme". D'où l'intérêt de la combinaison avec le vote majoritaire et le vote majoritaire pondéré; traiter le problème de non cohérence entre les classes et profiter des points forts de chaque classifieur.
3. Et en ce qui concerne la classe "Globule rouge", nous avons enregistré des bons taux de reconnaissances. Mais en terme de précision, les pourcentages ont légèrement baissé (dû aux Faux positifs). Et aussi, puisque nos images contiennent des globules rouges de type «Erythroblast (globule rouge avec un noyau)», toutes les segmentations faites ont détecté le noyau du globule rouge comme étant un fond.

Les taux de reconnaissances, les précisions et les résultats visuels (figures IV.8 et IV.9) montrent que la fusion permet d'améliorer sensiblement la qualité de la segmentation basée sur un processus de classification de pixels.

En général, nous avons noté une augmentation des pourcentages pour les différentes classes par rapport aux résultats obtenus sur la même base d'images [baghli2011]. La qualité de la segmentation d'un point de vue cytologique est grandement améliorée.

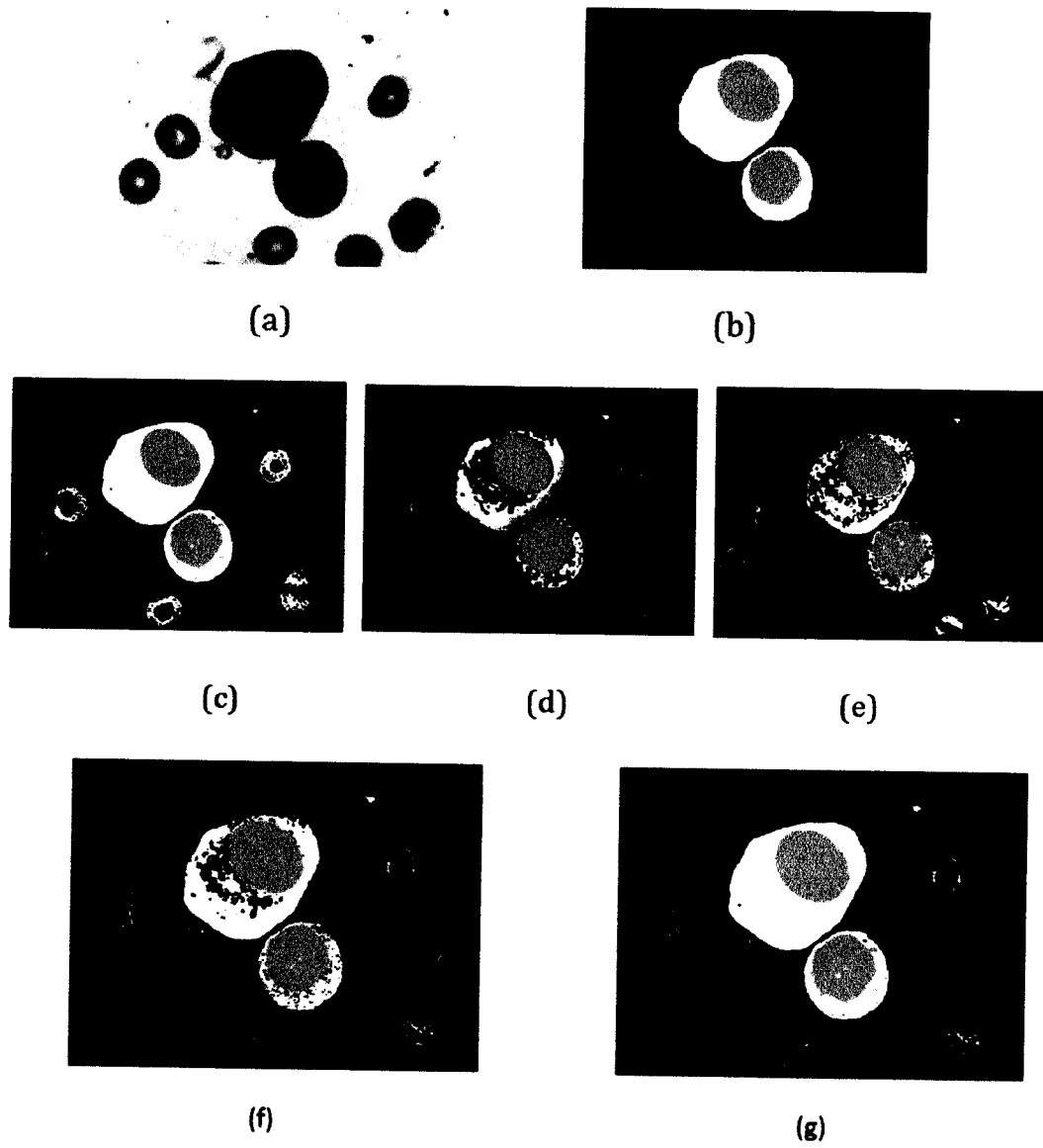
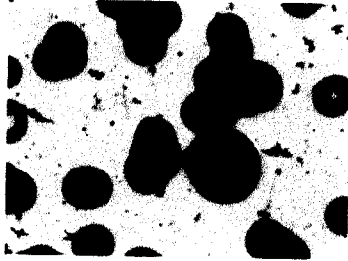


Figure IV.8: Image originale (a) Image étiquetée (b) Résultat du classifieur PMC(c) Résultat du classifieur SVM(d) résultat du classifieur KPPV(e) résultat du vote majoritaire (f) résultat du vote majoritaire pondéré (g)

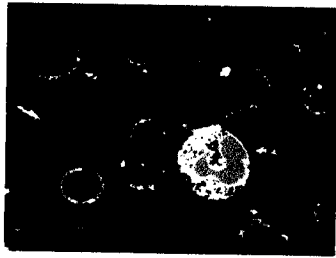
*Chapitre IV: Application*



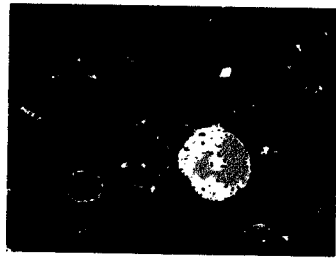
(a)



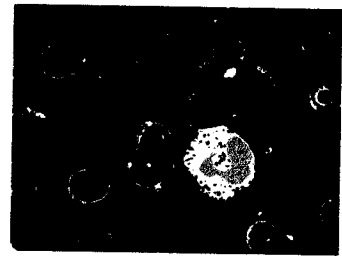
(b)



(c)



(d)



(e)



### **Conclusion générale et perspectives**

Dans cette étude nous nous sommes intéressés à développer une stratégie de segmentation en 4 classes d'images microscopiques pouvant contribuer à l'aide de décision du cytopathologiste afin d'améliorer le contrôle qualité en imagerie cellulaire. Pour discriminer les différentes classes en présence, nous avons proposé de considérer une approche multi-espaces couleur qui consiste à représenter les pixels dans plusieurs espaces couleur.

Notre objectif repose sur l'utilisation de plusieurs classifieurs entraînant des conflits de décisions, nous avons exploité la complémentarité qu'il peut exister entre eux et nous avons proposé deux méthodes de combinaison des classifications pixellaires.

Les résultats ont permis de montrer que l'utilisation de la combinaison des trois classifieurs (SVM, PMC et KPPV) en tenant compte de l'information de cinq espaces couleur, améliore les résultats de la segmentation finale.

Cette étude a permis de mettre en évidence que le choix des attributs à considérer est un des points délicats dans l'élaboration d'une méthode de classification. Dans notre cas, ce choix dépend des différents espaces couleurs ; RGB, LUV, HSL, YUV et HSV.

Le meilleur taux de reconnaissance obtenu (94.98 % pour le vote pondéré) est satisfaisant, mais comme dans tout travail de recherche, des améliorations restent toujours envisageables.

Nous pouvons proposer, comme perspectives, d'exploiter l'information du voisinage, d'enrichir les attributs (au lieu de s'appuyer seulement sur la couleur) et/ou encore procéder à d'autres types de segmentations. Aussi, envisager de passer à la caractérisation des cellules. Cette dernière s'intéresse à détecter le type du globule blanc pour diagnostiquer une maladie bien précise.