



Université ABBES LAGHROUR Khenchela  
Faculté des Sciences et de la Technologie  
Département de Génie Industriel  
جامعة عباس لغزور خنشلة  
كلية العلوم والتكنولوجيا  
قسم الهندسة الصناعية



N° Série : .....

## Mémoire de fin d'étude

*Pour l'obtention du diplôme de Master*

**Filière : Télécommunications**

**Spécialité : Systèmes des Télécommunications**

### THEME

***Elaboration des techniques basées sur  
les outils de l'intelligence artificielle  
pour la classification des images***

*Réalisé par : - Sara EL BAH*

***Soutenu le 22 / 11 /2020 Devant le jury composé de :***

***Mr.***

***Mr. Ali-maalem***

***Mr.***

***Président***

***Encadreur***

***Examinatrice***

***Université Abbes Laghrour-Khenchela***

***Université Abbes Laghrour-Khenchela***

***Université Abbes Laghrour-Khenchela***

*Promotion 2019/2020*

# Remerciements

Tout d'abord je présente ma gratitude à dieu tout puissant, pour m'avoir accordé l'effort, la Volonté et la patience de terminer ce travail.

Et je voudrais en premier lieu remercier mes parents.

Ensuite, j'aimerai bien remercier très sincèrement le Dr.ali maalem mon promoteur de ce travail, pour leurs conseils pertinents, et leurs orientations judicieuses et leur patience et diligence.

Mes remerciements vont aussi à l'ensemble des enseignants qui ont contribué durant ma Formation.

# Dédicace

Louage à dieu, Seigneur des mondes ;

C'est toi que nous adorons et de toi que nous implorons secours ;

Que la paix et la bénédiction soit sur son dernier envoyé ;

A mes chers parents qui ont été présents, à chaque instant, leur

Irremplaçable et inconditionnel soutien m'a permis d'écarter

les doute,

De soigner les blessures et partager mes joies ;

A ma petite famille, mes frères et sœurs ;

A tous mes tantes et mes oncles ;

A mes cousins ; A tout la famille El bah ; A tous mes amis sans

exception ;

# LISTE DES TABLEAUX

|  |   |
|--|---|
| Tableau 3.1 Loi de hebb.....   | 8 |
| Tableau 3.2 Base d'exemples d'apprentissage pour la loi de Hebb..... | 8 |
| Tableau 4.1 Taux de reconnaissance des approches.....                | 8 |

# LISTE DES FIGURES

|   |    |
|---|----|
| Figure I.1 - Exemples de courbes de transfert.....  | 5  |
| Figure I.2 - exemple d'histogrammes (nombre de points en fonction de l'intensité).....                | 6  |
| Figure I.3 Erosion de X par B5.2.....   | 9  |
| Figure I.4 Dilatation de X par B.....   | 9  |
| Figure II.1 Coupe d'une IRM mettant en évidence les différents tissus cérébraux.....                  | 15 |
| Figure II.2 illustre une coupe d'IRM cérébrale.....   | 16 |
| Figure II.3 Structure d'un neurone artificiel.....  | 16 |
| Figure II.4 Structure d'un neurone artificiel.....  | 25 |
| Figure II.5 Images synthétiques avant la segmentation utilisées pour les tests Image synthétique..... | 25 |
| Figure II.6 Résultats de la segmentation de l'image synthétique 1 par les deux méthodes.....          | 25 |
| Figure II.7 Résultats de la segmentation de l'image synthétique 2 par les deux méthodes.....          | 27 |
| Figure II.8 Résultats de la segmentation par les K-means pour $K = 4, K = 5, K = 6$ .....             | 28 |
| Figure II.9 Résultats de la segmentation par les FCM pour $K = 4, K = 5, K = 6$ .....                 | 28 |
| Figure III. 1 Neurone biologique / neurone artificiel.....  | 33 |
| Figure III. 2 Différents types de fonctions de transfert pour le neurone artificiel.....              | 33 |
| Figure III. 3 Perceptron mono-couche à $n = 4$ entrées et $p = 5$ sorties.....                        | 34 |
| Figure III. 4 Comportement en phase de reconnaissance d'un réseau de neurone multicouche.....         | 36 |
| Figure IV.1 : Image IRM cérébrale originale sans bruit.....   | 43 |
| Figure IV.2 : Image IRM cérébrale originale avec bruit gaussien.....                                  | 44 |
| Figure IV.3 : Résultats de segmentation obtenus par l'approche K-means en absence du bruit.....       | 45 |
| Figure IV.4 : Classes de l'image IRM cérébrale originales.....  | 45 |

|  |    |
|--|----|
| Figure IV.5: Classes de l'image IRM par l'approche K-means sans bruit.....   | 46 |
| Figure IV.6: Résultats de segmentation par l'approche K-means en présence du bruit gaussien....                      | 46 |
| Figure IV.7: Classes de l'image IRM par l'approche K-means avec bruit gaussien 5% .....                              | 47 |
| Figure IV.8: Résultats de segmentation par l'approche K-means en présence du bruit gaussien....                      | 47 |
| Figure IV.9: Classes de l'image IRM par l'approche K-means avec bruit gaussien.....                                  | 48 |
| Figure IV.10 : Résultats de segmentation obtenus par l'approche FCM en absence du bruit.....                         | 49 |
| Figure IV.11: Classes de l'image IRM par l'approche FCM sans bruit.....  | 50 |
| Figure IV.12: Résultats de segmentation par l'approche FCM en présence du bruit gaussien.....                        | 50 |
| Figure IV.13: Classes de l'image IRM par l'approche FCM avec bruit gaussien 5% .....                                 | 51 |
| Figure IV.14: Résultats de segmentation par l'approche FCM en présence du bruit gaussien.....                        | 51 |
| Figure IV.15: Classes de l'image IRM par l'approche FCM avec bruit gaussien 10% .....                                | 52 |
| Figure IV.16 : Résultats de segmentation obtenus par l'approche neuronale RBF.....                                   | 54 |
| Figure IV.17: Classification par l'approche neuronale RBF sans bruit (images binaires) .....                         | 55 |
| Figure IV.18: Résultats de segmentation par l'approche neuronale RBF en présence du bruit<br>gaussien 5% .....       | 55 |
| Figure IV.19: Classes de l'image IRM par l'approche neuronale RBF avec bruit gaussien 5%<br>(images binaires) .....  | 56 |
| Figure IV.20: Résultats de segmentation par l'approche neuronale RBF en présence du bruit<br>gaussien 10%.....       | 56 |
| Figure IV.21: Classes de l'image IRM par l'approche neuronale RBF avec bruit gaussien 10%<br>(images binaires) ..... | 57 |

# TABLES DE MATIERES

|                                    |    |
|------------------------------------|----|
| <b>Introduction Générale</b> ..... | 02 |
|------------------------------------|----|

## **Chapitre I traitement d'image**

|   |    |
|---|----|
| 1. Introduction .....                   | 04 |
| 2. Formation de l'image numérique ..... | 04 |
| 3. Amélioration de l'image.....         | 04 |
| 3.1 Traitement point à point .....      | 05 |
| 3.2 Méthodes locales .....              | 06 |
| 4. Binarisation des images.....         | 07 |
| 5. Morphologie mathématique .....       | 08 |
| 6 Segmentation .....                    | 10 |
| 7 . Conclusion.....                     | 12 |

## **Chapitre II classification des images**

|   |    |
|---|----|
| 1.Introduction.....                     | 14 |
| 2. La segmentation des images.....      | 15 |
| 3.Segmentation par classification ..... | 19 |
| 4. L'algorithme des K-means.....        | 22 |
| 5. L'algorithme des FCM5.....           | 22 |
| 6 Résultats et discussions.....         | 24 |
| 7. Conclusion.....                      | 28 |

## **Chapitre III réseaux des neurones**

|   |    |
|---|----|
| 1. Introduction.....                            | 30 |
| 2. le model neurophysiologique.....             | 31 |
| 3 modélisations mathématiques.....              | 32 |
| 4. Fonctionnement.....                          | 34 |
| 5.comportements combinatoire et séquentiel..... | 35 |
| 6. Mémoires associatives.....                   | 41 |
| 7. conclusion.....                              | 41 |

## **Chapitre 4 :Application d'une approche neuronale pour la segmentation des images médicales**

|  |    |
|--|----|
| IV.1. Introduction.....                      | 43 |
| IV.2. Techniques proposées.....              | 43 |
| IV.2.1. Algorithme K-means.....              | 44 |
| IV.2.2. Algorithme FCM (Fuzzy C-Means) ..... | 48 |

|                                     |           |
|-------------------------------------|-----------|
| IV.2.3. Approche neuronale RBF..... | 52        |
| IV.2.4. Critère d'évaluation.....   | 57        |
| IV.3. Conclusion.....               | 58        |
| <b>Conclusion Générale.....</b>     | <b>60</b> |
| <b>Bibliographie.....</b>           | <b>62</b> |

---

---

# ***Introduction Générale***

---

---

## INTRODUCTION GENERALE

Les documents anciens représentent une richesse culturelle et scientifique inestimable. La sauvegarde de ce patrimoine qui souffre des outrages du temps et de la vicissitude des hommes, interpelle tant les scientifiques que les politiques pour préserver ces trésors des risques de détérioration voire de disparition. Le potentiel véhiculé aujourd'hui par les technologies de l'information et de la communication, la maturité des systèmes de traitement d'images, la généralisation du web à haut débit, marquent un dans la conservation et l'exploitation de ces documents. Le défi consiste à préserver ces fonds par une numérisation massive et une série de traitements appropriés à l'aide d'algorithmes fiables et performants. L'hétérogénéité des pages, la dégradation de nombreux documents, sont des exemples reflétant la spécificité et les enjeux scientifiques que les chercheurs doivent surmonter.

Etroitement corrélé à cette thématique se pose le problème délicat de l'indexation des ces documents. Cette opération consiste à interpréter et reformuler un contenu documentaire pour le rendre accessible et exploitable par un système d'information. Cette opération d'indexation des documents anciens doit tenir compte de leurs formes, la mise en page, la variabilité de style et de fontes souvent complexes, l'absence de structure logique, la non homogénéité des caractères, mots et graphes souvent inconstants, l'inclinaison des textes etc. Différentes méthodes sont exploitées donnant des résultats souvent mitigés, à titre d'exemple, l'approche structurale construite à partir d'une étude des caractéristiques structurales des documents anciens opère sans connaissance préalable du modèle de document et donne des taux de reconnaissance élevés, mais elle est lente et sensible aux inclinaisons. L'approche qui semble la plus appropriée à notre avis est celle basée sur la comparaison des contours des mots. D'où notre proposition s'une approche semi-automatique d'indexation et de recherche documentaire, où le choix des indexes se fait manuellement, tandis que leurs paramétrage (détection des primitives de bas niveau) et le stockage se fait automatiquement.

Donc l'idée de base de notre approche est la recherche des indexes, détection de leurs contours, paramétrage des contours et stockage dans une base de données afin d'effectuer des recherches ultérieures. Le paramétrage des contours est effectué sous forme de chaîne de codes, tandis que la procédure d'optimisation permettra de résoudre partiellement le problème de taille des mêmes manuscrits ainsi que l'aspect des rotations.

---

---

# *CHAPITRE I:*

*NOTION SUR LE TRAITEMENT DES IMAGES*

---

---

## **CHAPITRE I**

### **NOTION SUR LE TRAITEMENT DES IMAGES**

#### **1.Introduction**

Tout système de traitement d'image peut être vu comme une combinaison de deux phases, L'acquisition et le traitement proprement dit. La qualité des résultats dépend de l'algorithme mis en place et de son adéquation au problème posé, et aussi, de la qualité initiale des images traitées En informatique, l'image est représentée sous format numérique. Une image «réelle » va être transformée en une image numérique par différents outils de transformation (caméra, scanner, satellite....).

#### **2. Formation de l'image numérique**

Une image est représentée par une matrice de dimension « nombre de ligne » x « nombre de Colonnes ». Chaque élément de la matrice, nommé pixel, représente l'intensité lumineuse comprise entre 0 et 255, soit 256 niveau de gris, le niveau de gris 0 représente le noir tandis que le niveau de gris 255 représente le blanc. Autrement dit, une image est une forme discrète d'un phénomène continu obtenue après discrétisation. Cette forme est bidimensionnelle et les informations qui la présentent définissent les intensités lumineuses (couleurs ou niveaux de gris  $I : [0, L-1] \times [0, C-1]$  définit une image de L lignes et C colonnes dont l'information portée est défini dans un espace à p dimensions ; Si I est une image en niveaux de gris, alors  $p = 1$ . Si I est une image couleur, alors  $p = 3$ .

#### **3. Amélioration de l'image**

Supposons que l'on dispose d'une image numérisée qui contient des escaliers. Peut-on améliorer cette image par filtrage? Remarquons d'abord que toute image numérisée correspond à un signal: il se peut que l'effet d'escalier corresponde à la réalité du signal, et non pas à un effet de lissage. Toute technique de filtrage qui travaille à partir de l'image numérisée peut améliorer l'aspect visuel de l'image ou le détériorer, selon les caractéristiques originales de l'image.

### 3.1 Traitement point à point

La technique de traitement la plus simple consiste à appliquer à chaque pixel de l'image une fonction qui transforme son intensité. Cette fonction peut être représentée par une courbe de transfert (Figure I.1) qui donne en ordonnée l'image de chaque niveau d'intensité en abscisse.

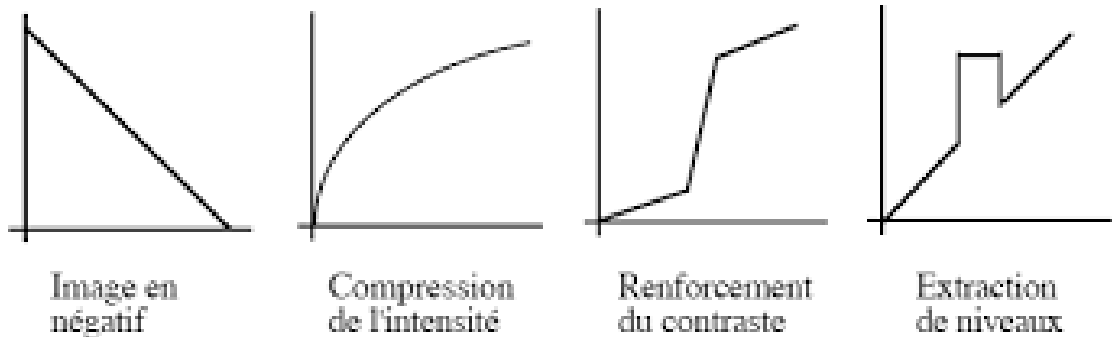


Figure I.1 - Exemples de courbes de transfert

La compression de l'intensité est utile pour des images qui ont une très large plage d'intensité, comme par exemple une transformée de Fourier. Le renforcement du contraste permet d'augmenter artificiellement le contraste d'une image terne. L'extraction de niveaux permet de mettre en évidence les parties de l'image qui représente une intensité donnée. Pour déterminer la courbe de transfert la mieux adaptée à un traitement donné il faut calculer l'histogramme de l'image qui représente l'ensemble des pixels en fonction de l'intensité. La technique d'égalisation de l'histogramme consiste à transformer l'image de telle sorte que l'histogramme soit plus plat, c'est-à-dire, il s'agit de rendre la fonction de densité de probabilité aussi uniforme que possible (Figure I.2). Pour obtenir ce résultat, il suffit de prendre comme fonction de transfert la fonction  $f$  suivante

$f(i) = (1/n) \text{ Somme } i=0..i H(i)$  où:  $n$  est le nombre total de pixels,  $H(i)$  le nombre de pixels d'intensité  $i$ , fourni par l'histogramme.

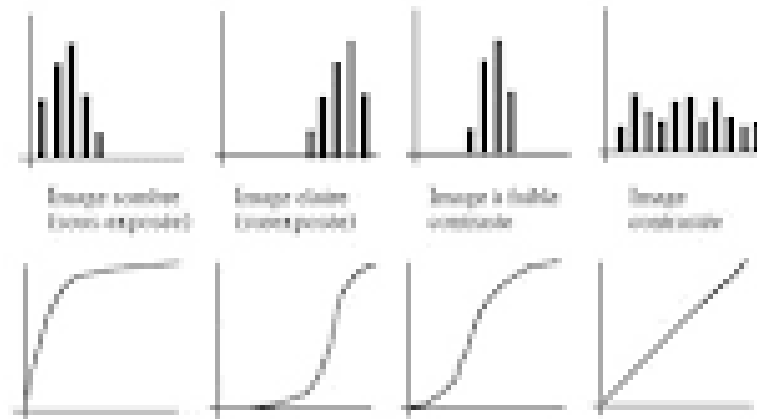


Figure I.2 - exemple d'histogrammes (nombre de points en fonction de l'intensité)

### 3.2 Méthodes locales

Les méthodes locales utilisent des techniques voisines des méthodes globales, mais en l'appliquant à un voisinage de chaque pixel. Ainsi, pour améliorer le contraste de l'image localement, on peut appliquer la technique d'égalisation d'histogramme à chaque pixel en utilisant les valeurs des pixels voisins. Pour chaque pixel de l'image on calcule l'histogramme sur un voisinage (par exemple  $7 \times 7$ ). On calcule aussi la fonction de transfert par égalisation de l'histogramme en appliquant cette fonction au pixel central. Le calcul de l'histogramme peut être optimisé le long d'une ligne puisqu'à chaque déplacement, 7 pixels sortent du voisinage et 7 pixels entrent. De plus, il suffit de connaître la valeur de la fonction de transfert pour le pixel central et donc calculer la somme des  $7 \times 7 / 2$  premiers histogrammes.

Une autre façon d'améliorer le contraste est de calculer la moyenne  $m$  et l'écart type  $s$  de la Zone de l'image autour de chaque pixel et d'appliquer la transformation suivante :

$$p' = A(p - m) + m \quad A = k M / s$$

Où:  $p$  est l'intensité du pixel,  $p'$  l'intensité après transformation,  $M$  la moyenne globale des Intensités et  $k$  un réel entre 0 et 1  $A$  étant inversement proportionnel à la variance autour du pixel considéré, il est plus grand dans les zones de faible contraste. Le terme  $A(p-m)$  renforce donc le contraste dans ces zones. En pratique, il est parfois nécessaire de borner  $A$  entre deux valeurs  $A_{min}$  et  $A_{max}$ .

### 4 Binarisation des images

La binarisation ou le seuillage est la technique de classification la plus simple où les pixels de l'image sont partagés par un seul seuil  $s$  en deux classes : ceux qui appartiennent au

fond et ceux qui appartiennent à l'objet. L'image est alors séparée en deux classes de façon à ce que l'information comprise entre 0 et s est retenue et l'autre non, ou vice-versa.

Soit l'image I (M x N), supposons que  $f(x, y)$  représente le niveau de gris du pixel aux coordonnées (x, y), s est le seuil choisi, les pixels de l'objet sont ceux ayant le niveau de gris

Inférieur à s et les autres ayant le niveau de gris supérieur à s sont des pixels du fond.

Alors,

L'image binarisme G est déterminée par les pixels (x, y) dont la valeur est :

$$g(x,y) = \begin{cases} 1 & \text{si } f(x,y) > s \\ 0 & \text{si } f(x,y) \leq s \end{cases} \quad (1)$$

Il existe trois grandes techniques de sélection du seuil s :

Global, local et adaptatif [OTS79]. Comme il y a des différentes façons de déterminer le seuil s, il peut être considéré comme une fonction sous forme de  $s = t(x, y)$ ,  $p(x, y)$ ,  $f(x, y)$  où  $p(x, y)$  représente des propriétés locales du point (x, y). Si s ne dépend que de la valeur  $f(x, y)$  du point, le seuil est *global*, s'il dépend en plus de  $p(x, y)$ , s est un seuil *local*. Et si s dépend à la fois de (x, y), de  $p(x, y)$  et de  $f(x, y)$ , on dit un seuil adaptatif ou dynamique.

Parmi les méthodes de binarisation les plus utilisées dans le traitement des documents historiques nous avons:

#### 4.1 Binarisation efficace des documents historiques :

Cette méthode combine plusieurs méthodes de binarisation d'où  $R_1(x, y)$ ,  $R_2(x, y), \dots, R_n(x, y)$  représentent leurs résultats après une application sur l'image  $IF(x, y)$ , n est un nombre impair,  $n=2m+1$ . [GAT08]

Chaque image binarisée  $R_i$  est définie par :

$$R_i(x, y) = \begin{cases} 1 & \text{le trait} \\ 0 & \text{le fond} \end{cases} \quad (2)$$

## 4.2 . Méthode à seuillage adaptatif

Cette méthode est à seuillage adaptatif, et appliquée à des documents, au niveau de gris

et de très faible qualité. Le seuil local est calculé en fonction de la moyenne locale « moy », l'écart type et le niveau de gris minimum local.

Le seuil est calculé comme suit :

$$1. \text{Calculer la moyenne "moy" pour chaque pixel : } moy = \frac{\sum_{i=1}^N I_i}{N} \quad (3)$$

Où N est le nombre de pixel dans un voisinage local et  $I_i$  la valeur de l'intensité du pixel  $i$ .

$$2- \text{ Calculer l'écart type } s : s = \sqrt{(1/N) \times \sum (I_i - moy)^2} \quad (4)$$

3- Calculer le niveau de gris minimum M

4- Calculer  $R_s$  est le rang dynamique de l'écart type du niveau de gris local.

5- le seuil T est donné par :

$$T = (1-a_1) \times moy + k_1 \times (s/R_s)^3 \times (moy - M) + k_2 \times (s/R_s)^2 \times M \quad (6)$$

Où:  $0.15 \leq a_1 \leq 0.25$ ,  $0.05 \leq k_1 \leq 0.15$  et  $0.01 \leq k_2 \leq 0.05$ .

## 5 Morphologie mathématique

### 5.1 Erosion :

Cette opération est très importante avec la dilatation. Son principe est le Suivant : soit un ensemble B, repéré par son centre p, B appartient à l'ensemble X et Représente une forme quelconque. B est déplacé de telle sorte que son centre p occupe toutes les positions de l'espace. Est ce que l'ensemble B est entièrement inclus dans X pour chaque position de p ? Donc l'ensemble des p répondant à cette question forme une érosion de X par

B notée :  $X \ominus B$ .

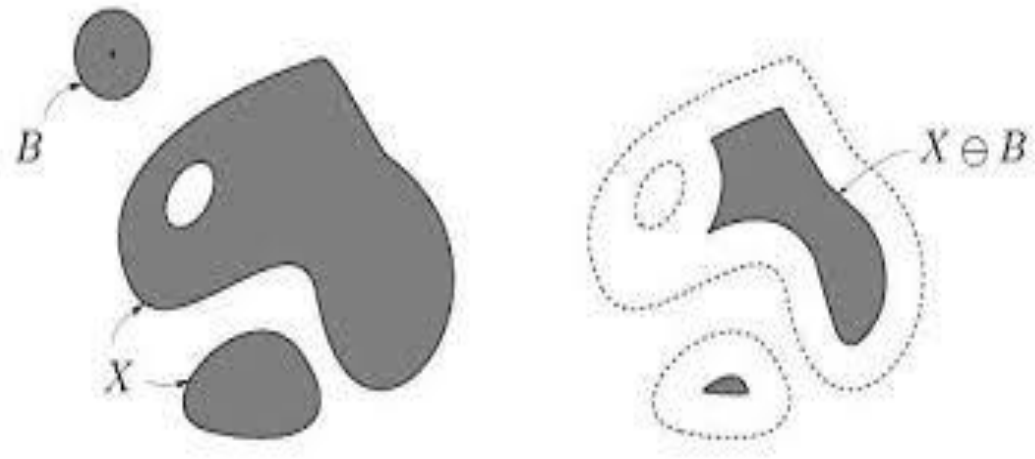


Figure I.3 Erosion de X par B

### 5.2 Dilatation :

L'opération de dilatation se définit d'une manière analogue, on obtient le dilaté de X par B en prenant la totalité de la surface de couverture lorsque l'origine de B occupe successivement toutes les positions dans X. Elle est notée par  $X \oplus B$

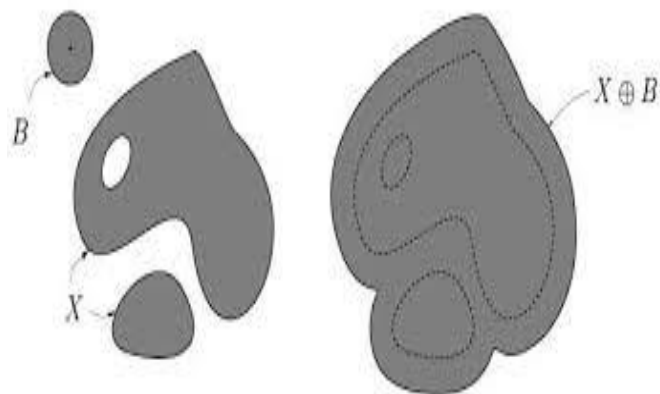


Figure I.4 Dilatation de X par B

### 5.3 L'ouverture:

Les opérations "Erosion" et "Dilatations" peuvent être combinées et une Opération d'érosion suivie d'une dilatation s'appelle une ouverture et notée par  $X \circ B$

Où :  $X \circ B = (X \ominus B) \text{ xor } B$

### 5.4 La fermeture :

L'opération de fermeture est obtenue en inversant l'ordre des opérations utilisées pour définir l'ouverture, elle notée par  $X \bullet B$ , Où :  $X \bullet B = (X \text{ xor } B) \ominus B$ .

## 6 Segmentation :

La segmentation des images consiste à regrouper les pixels qui ont les mêmes propriétés. Afin de former des régions connexes. Il existe deux familles d'approches que l'on

Peut faire coopérer : les approches contours et les approches régions. Dans les approches Contours les régions sont déterminées par leurs contours par contre dans les approches régions .D'autres critères font intervenir tels que l'agrégation de pixels et l'homogénéité.

### 6.1 Segmentation contour :

L'extraction du contour consiste à rechercher les points de contour, réduire le contour à un pixel, fermer les contours ouverts, et enfin, coder le contour.

### 6.2 L'approche gradient :

Consiste à calculer la dérivée première en point de coordonnées p,q Comme suit :



### 6.3 L'approche Laplacien:

Ces méthodes ont été proposées en 1976. Elles ont eu une grande Importance historique, étant considérées comme le prototype du détecteur de contour inspiré Des systèmes biologiques (*primalsketch* de Marr) Elles utilisent le fait que le Passage par zéro du Laplacien permet de bien mettre en évidence les extremums de la dérivée :

$$\frac{\partial^2 f}{\partial^2 x} \Big|_{P,Q} = 1/4\lambda x^2 [f(p+1, q) - 2 f(p,q) + f(p-1,q)] \quad (10)$$

$$\frac{\partial^2 f}{\partial^2 y} \Big|_{p,q} = 1/4\lambda y^2 [f(q+1, p) - 2 f(p,q) + f(p,q-1)] \quad (11)$$

Ces formules peuvent être transformées en opérations matricielles en utilisant l'un des deux Masques de convolution suivants :

|    |    |    |
|----|----|----|
| 0  | -1 | 0  |
| -1 | 4  | -1 |
| 0  | -1 | 0  |

|   |   |   |   |
|---|---|---|---|
| 1 | - | 1 | - |
| 1 | - | 8 | - |
| 1 | - | 1 | - |

### 7. Conclusion :

Dans ce chapitre nous avons présenté quelques notions sur le traitement de l'image, qui est un domaine très vaste. Pour cela nous avons expliqué les principes des méthodes qui doivent être appliquées sur les images avant de procéder à l'indexation, notamment : l'amélioration de l'image, la binarisation et la Segmentation. Cette panoplie de méthodes nous offre la possibilité d'utiliser celles qui sont plus efficaces et faciles.

---

---

## *CHAPITRE II:*

### *Classification des images médicales*

---

---

## Chapitre II

### Classification des images médicales

#### II.1 Introduction

L'une des étapes critiques du traitement d'images est la segmentation, celle-ci consiste à localiser dans une image les régions (ensembles de pixels) appartenant à une même structure (objets ou scène imagés). Si l'homme sait naturellement séparer des objets dans une image c'est grâce à des connaissances de haut niveau (compréhension des objets et de la scène). Mettre au point des algorithmes de segmentation de haut niveau (chaque région est un objet sémantique) est encore un des thèmes de recherche les plus courants en traitement d'images.

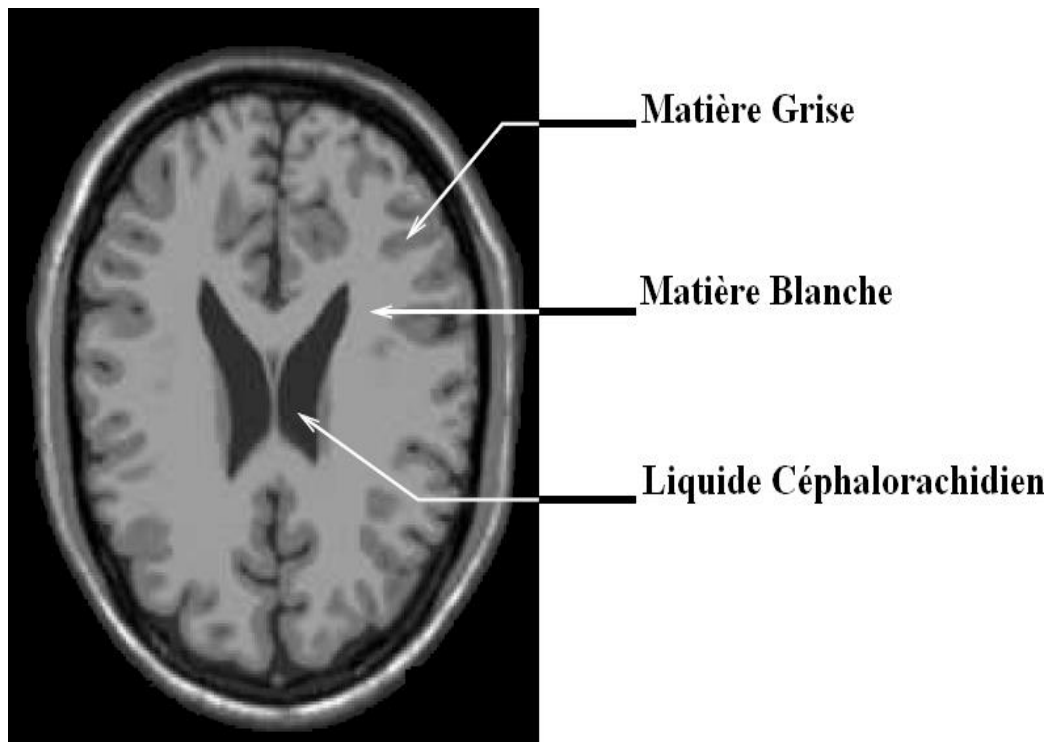
La segmentation est à la base de nombreuses applications tant en vision industrielle, qu'en imagerie médicale. De nombreuses recherches ont eu lieu sur les méthodes de segmentation. Il en résulte un très grand nombre de méthodes dont la comparaison, soit en termes de structure soit en termes de performance, est très difficile.

Grace à l'évolution technologique dans le domaine médicale, un grand nombre de modalités d'acquisition d'images médicales a vu le jour (Scanner, IRM, TEP,...etc.). Ces technologies ont grandement augmenté nos connaissances en matière d'anatomie du corps et jouent actuellement un rôle prépondérant dans le diagnostic médical. Cependant, la croissance du nombre de modalité d'acquisition confronte l'expert aujourd'hui à un volume très important d'information à traiter et diagnostiquer, ainsi, afin d'assister l'expert lors de son analyse, il est devenu primordial de faire appel à des outils informatiques, et plus particulièrement aux algorithmes de segmentation d'images qui consiste à délimiter les différentes structures, pathologies ou région d'intérêt figurant dans l'image.

Ce chapitre est consacré à la segmentation d'images médicales. Nous aborderons d'abord quelques notions fondamentales relatives à la segmentation d'images, nous verrons par la suite les différentes techniques et approches de segmentation. Cependant, sont regroupées en trois grandes catégories de segmentation : la segmentation par régions, la segmentation par contours et la segmentation par classification. Nous présenterons enfin le principe de notre approche de segmentation.

## 2 La segmentation des images

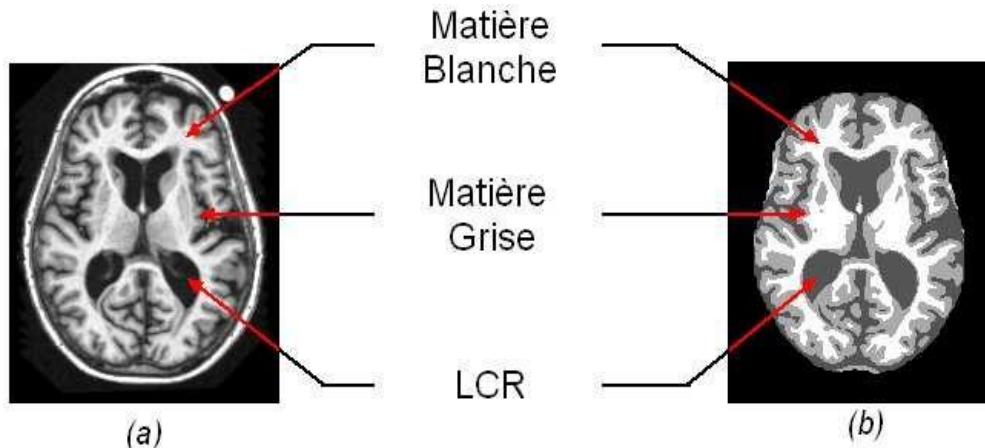
La segmentation d'images est un domaine de recherche en pleine activité qui couvre un champ d'application très vaste (Imagerie médicale, Robotique, Imagerie Satellitaire...etc.). En imagerie médicale la segmentation d'organe est si répandue qu'il serait difficile d'énumérer la liste des organes communément segmentés, mais en général la liste inclurait au moins ce qui suit : Le cerveau, le cœur, le foie et les vaisseaux sanguins. Notons cependant qu'il n'existe actuellement pas d'algorithme universel de segmentation applicable à tout type d'image; en effet le résultat de la segmentation dépend fortement de l'information sémantique à discerner, celle-ci varie d'une application à une autre [54] ; de ce fait nous nous focaliserons essentiellement sur la segmentation d'images médicales et plus particulièrement aux images cérébrales (figure II.1).



*Figure II.1 : Coupe d'une IRM mettant en évidence les différents tissus cérébraux .*

La segmentation d'image médicale consiste à associer à chaque voxel de l'image médicale une étiquette ou un label indiquant ainsi le type de tissu ou la structure anatomique, la collection d'étiquettes produite à travers le processus de segmentation s'appelle la carte d'étiquettes (La

bel map). Celle-ci décrit la correspondance spatiale entre les intensités des voxel dans l'image et les différents types de structure anatomiques présents dans l'image.



La figure II.2 illustre une coupe d'IRM cérébrale (a) ainsi que sa segmentation en tissus (b).

### 2.1 Définition :

Il n'est pas aisé de trouver une seule définition de la segmentation d'image car cette tâche est souvent confondue avec la classification. "La segmentation est un traitement de bas niveau qui consiste à créer une partition de l'image  $A$  en sous-ensembles  $R_i$ , appelés régions tels qu'aucune région ne soit vide, l'intersection entre deux régions soit vide et l'ensemble des régions recouvre toute l'image. Une région est un ensemble de pixels connexes ayant des propriétés communes qui les différencient des pixels des régions voisines".

On peut dire que :

La segmentation est la partition d'une image en un ensemble de régions qui ne se chevauchent pas et dont l'union est l'image entière. Quelques règles à suivre pour obtenir une segmentation sont :

- Les régions doivent être uniformes et homogènes par rapport à certaines caractéristiques (niveau de gris, écart type, gradient).
- Leurs intérieurs doivent être simple et sans beaucoup de petits trous (des parties de région non segmentés).
- Les régions adjacentes doivent avoir des valeurs très différentes par rapport à la caractéristique prise en compte dans la segmentation.
- Les limites de chaque région doivent être simples et spatialement précises.

En termes mathématiques :

$$\begin{cases} \bigcup_{i=1}^n R_i = I \\ R_i \neq \emptyset \quad \forall i = 1 \dots n \\ R_i \cap R_j = \emptyset \quad \forall i, j \text{ Avec } i \neq j \end{cases} \quad (2.1)$$

Où  $R_i$  est l'ensemble des régions formant l'image  $I$ .

Dans un contexte médical, les régions  $R_i$  correspondent aux différentes structures anatomiques constituant les régions d'intérêt. La détermination automatique du nombre de région  $N$  reste néanmoins une vraie problématique.

Le prédicat  $P$  est utilisé pour tester l'uniformité des ensembles  $R_i$ . Ces sous-ensembles constituent les régions de l'image. Une segmentation de l'image est donc sa décomposition en un ensemble de région uniforme. Le critère d'uniformité restant à déterminer. La première condition implique que tout pixel ou voxel de l'image appartient à une et une seule région. En d'autre terme cela signifie que l'algorithme de segmentation ne doit pas se terminer avant d'avoir traité tous les points. La seconde condition signifie que les régions doivent être disjointes, Il n'y a donc aucun chevauchement ou intersection entre régions. La troisième condition implique que chaque région est uniforme, selon le prédicat d'uniformité défini au préalable. Enfin, la quatrième et dernière condition est une condition de maximalité indiquant que la fusion de deux régions adjacentes ne doit pas être uniforme.

Le prédicat d'uniformité  $P$  est à la base de la définition des régions, il doit être choisi à travers des descripteurs de pixel susceptibles de permettre une bonne identification des objets. Parmi ceux-ci on peut citer : le niveau de gris, la couleur, la texture, la géométrie, et d'autres paramètres pertinents.

## 2.2 Le choix d'une technique de segmentation:

Le choix est lié à :

- La nature de l'image (éclairage, contours, texture, etc).
- Aux opérations en aval de la segmentation (compression, reconnaissance des formes, Mesures, etc).
- Aux primitives à extraire (droites, régions, textures, etc).

- Aux contraintes d'exploitation (temps réel, espace mémoire, etc) ;

### 2.3 Objectifs de la segmentation :

- Fournir des régions homogènes (selon un critère donné).
- Localiser de manière précise les contours des régions.
- L'étude et l'interprétation des structures anatomiques.
- Réduction de bruit.

### 2.4 Différentes approches de la segmentation

Le choix d'une méthode de segmentation est lié à plusieurs facteurs Comme : la nature de l'image, les conditions d'acquisition (bruit), les primitives à extraire (contours, textures, etc.) et bien évidemment les contraintes d'exploitation (fonctionnement en temps réel, type, mémoire vive et physique disponible).

En effet dans la littérature, plusieurs manières de catégoriser les méthodes de segmentation d'images ont été proposées classées en quatre classes :

- a. Approche par contours
- b. Approche par classification de Pixels
- c. Approche régions
- d. Approche hybride.

#### *A. Approche par contours*

De façon générale, un contour est défini comme étant la frontière entre deux régions. La détection de contours consiste à repérer les points d'une image numérique qui correspondent à un changement brutal de l'intensité lumineuse.

Les méthodes de détection de contours donnent de bons résultats quand les contours de l'image sont bien définis. Cependant, dans le cas des images bruitées ou faiblement contrastées, les méthodes contours nécessitent une étape supplémentaire afin de fermer les bords des régions.

#### *B. Approche régions*

Les approches régions ont pour but de mettre en évidence les régions homogènes de l'image. Il s'agit de rechercher des ensembles de pixels partageant des propriétés communes. Les régions sont différenciées entre elles par des propriétés élémentaires basées sur des critères locaux tels

que le niveau de gris de chaque pixels, ou bien sur un attribut estimé dans le voisinage du pixel tel que la valeur moyenne, la variance ou des paramètres de texture

### *C. Approche par classification de pixels*

Ce type de méthode considère une région comme un ensemble de pixels connexes appartenant à une même classe. Elles supposent donc que les pixels qui appartiennent à une même région possèdent des caractéristiques similaires et forment un nuage de points dans l'espace des attributs.

## **II.3. Segmentation par classification :**

### **II.3.1. Classification floue :**

Dans une méthode de classification floue, on affecte au pixel un degré d'appartenance pour chacune des classes de la partition qui indique la probabilité que le pixel y appartienne.

### **II.3.2. Classification dure :**

Dans une méthode de classification dure, un pixel ne peut être affecté qu'à une seule classe dans la partition.

#### *A. Classification supervisée :*

Dans la classification supervisée, le nombre de classes est connu et on dispose d'un ensemble de pixels déjà étiquetés, servant d'ensemble d'apprentissage. Il s'agit alors de pouvoir associer chaque nouveau pixel à la classe la plus adaptée en se servant des pixels déjà étiquetés. Permet les méthodes de cette classification :

*Les réseaux de neurones :* Un "neurone formel" est une fonction algébrique non linéaire et bornée, dont la valeur dépend de paramètres appelés coefficients ou poids. Les variables de cette fonctions ont habituellement appelées "entrées" du neurone, et la valeur de la fonction est appelée sa "sortie". La période d'apprentissage consiste à calculer les différents coefficients, de sorte à ce que les sorties du réseau soient égales aux valeurs souhaitées (à un seuil d'erreur près) pour les exemples utilisés

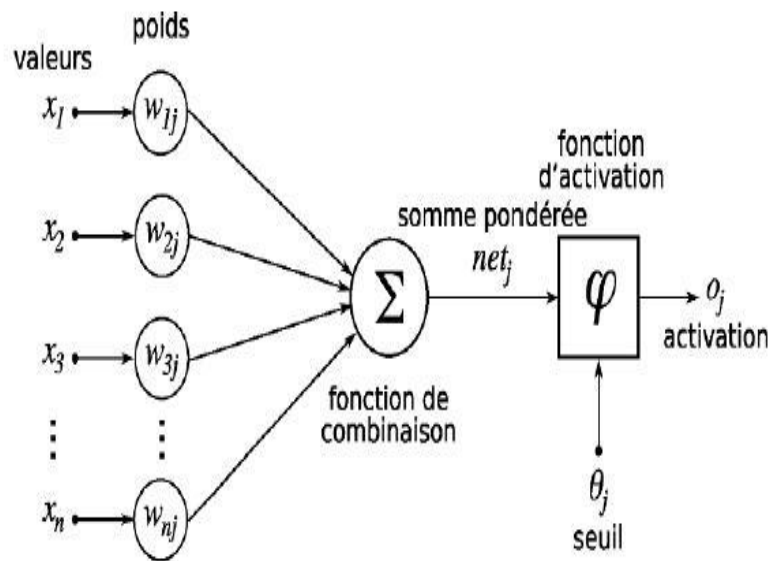


Figure II.3: Structure d'un neurone artificiel

### B. La classification non supervisée :

Dans la classification non supervisée, aucune information sur le nombre et le contenu des classes possible n'est fourni. L'objectif est alors de pouvoir regrouper automatiquement ceux des autres classes.

Permet les méthodes de cette classification :

#### B.1. La méthode *k*-moyennes (K-means) :

C'est l'un des algorithmes le plus connu et le plus utilisé en classification non supervisée du fait de sa simplicité de mise en œuvre. Il vise à produire un partitionnement des pixels de manière à ce que les pixels d'une même classe soient semblables et les pixels issus de deux classes différentes soient dissemblables. Chaque classe de la partition est définie par ses objets et son centriode.

Le principe de K-means est de choisir aléatoirement un ensemble de centres fixé a priori et de chercher itérativement la partition optimale. Chaque individu (également appelé centriode ou centroid en anglais) est associé au centre le plus proche, après l'affectation de toutes les données la moyenne de chaque classe est calculé, elle constitue les nouveaux

Représentants des groupes, lorsqu'ont abouti à un état stationnaire (aucune donnée ne change de groupe) l'algorithme est arrêté.

Le principe des K-means est de minimiser la fonction objectif :

$$J(x, v) = \sum_{i=1}^N \sum_{k=1}^C \|x_i - v_k\|^2 \quad (2.2)$$

$c$  est le nombre de classe.

$N$  est le nombre de données.

$n$  est le nombre de donnée de chaque classe.

$v_k$  est le centre de classe  $i$ .

$\|\cdot\|$  : est la norme Euclidienne qui mesure la similarité entre un pixel est un autre.

Tenant compte de la nouvelle fonction objectif (2.2), la mise à jour des centres des classes  $v$  se fait avec l'équation (2.3):

$$V_K = \frac{1}{n} \sum_{i=1}^n x_i \quad (2.3)$$

**B.2. Les C-moyennes floues ou Fuzzy C-Means(FCM) :**

a généralisé le critère  $J_2$  proposé par Dunn à toute une famille d'algorithmes qu'il a appelée C-moyennes floues. Le nombre de groupes est désormais désigné par C (et non plus K). L'algorithme des C-moyenne floues effectue une optimisation itérative en évaluant de façon approximative les minimums d'une fonction d'erreur. Il existe toute une famille de fonction d'erreur associées à cet algorithme qui se distinguent par des valeurs différentes prises par un paramètre réglable  $m$ , appelé indice de flou (fuzzy index) et qui détermine le degré de flou de la partition obtenue. Les FCM sont un cas particulier d'algorithmes basés sur la minimisation d'un critère ou d'une fonction objectif.

$$J_m(u(\cdot), v) = \sum_{j=1}^N \sum_{i=1}^C \mu_{ji}^m(x) \|x_j - v_i\|^2 \quad (2.4)$$

Avec :

$$U_{ij} = 1, U_{ij} \in [0.1], \quad 1 \leq j \leq N, 1 \leq i \leq C \quad (2.5)$$

Les centres et les degrés d'appartenance sont, pour une valeur de  $m$  donnée, calculés à l'aide des deux formules suivantes :

$$U_{ij} = \sum_{K=1}^C \left( \frac{\|x_j - v_i\|}{\|x_j - v_K\|} \right)^{\frac{-2}{m-1}} \quad (2.6)$$

$$V_i = \frac{\sum_{j=1}^n (U_{ij})^{m_{xj}}}{\sum_{j=1}^n (U_{ij})^{m_x}} \quad (2.7)$$

**4. L'algorithme des K-means**

L'algorithme suivant résume les étapes principales pour l'exécution

**Etape 1:** Choisir aléatoirement les C centres initiaux  $v_1, v_2, v_3, \dots, v_C$  à partir des points

$$\{x_1, x_2, \dots, x_N\}$$

**Etape 2 :** Assigner  $x_i$  à la classe  $k$  ;  $k = 1 \dots c$  si :

$$\|x_i - v_k\| \leq \|x_i - v_p\| ; p = 1, \dots, C \text{ Avec } i \neq p$$

**Etape 4 :** Répéter de l'étape 2 et 3 jusqu'à atteindre la convergence :  $v_k^* = v_k$   $k = 1, \dots, c$ .

#### 4.1. Contraintes liées à la classification par K-means

Cet algorithme a plusieurs inconvénients :

- La classification finale dépend du choix de la partition initiale et le choix aléatoire des centres initiaux.
- Convergence vers un minimum local.
- L'algorithme est fortement sensible au bruit.

Afin de remédier à ces inconvénients, Dunn propose une meilleure détection de classes compactes et bien séparées en élargissant le domaine de la recherche à l'ensemble des partitions floues de X. La généralisation consiste à minimiser le critère  $J_2$  [14] ci-dessous dans lequel le carré de la distance d'un élément  $x$  au centroïde  $v_i$  est pondéré par le carré du degré d'appartenance de  $x$  à la classe  $i$  :

$$J_2(u(\cdot), v) = \sum_{x \in X} \sum_{i=1}^c \mu_i^2(x) d^2(x, v_i) \quad (2.8)$$

Où  $(x)$  dénote ici la partition floue, définie par les différentes fonctions d'appartenance  $u_i$  des parties .

## 1. L'algorithme des FCM

Dans ce qui suit nous, présentons les étapes de l'algorithme FCM pour l'exécution :

**Etape 1 :** fixer les paramètres  $C$  : nombre de classe  $\varepsilon$ : seuil représentant l'erreur de convergence (par exemple  $\varepsilon = 0.001$ ),  $m$ : degré de flou, généralement  $m = 2$ .

**Etape 2 :** Initialiser la matrice de degrés d'appartenances  $U$  par des valeurs aléatoires dans l'intervalle  $[0, 1]$ .

**Etape 3 :** calculer le prototype  $v$  par la relation (6).

Calculer  $(u(\cdot), v)$  par la relation (4).

$$|J_m(u(\cdot), v) - J_m(u(\cdot), v)| < \varepsilon$$

### II.5.1 Contraintes liées à la classification

- Résultats de segmentation dépendant de l'étape d'initialisation.
- Blocage de la méthode dans des minima locaux.
- Les algorithmes FCM souffrent du manque de prise en compte d'informations spatiales et fournissent des classifications moins efficaces en présence de bruit.

### II.6 Résultats et discussions

Afin de retracer et comparer les différentes méthodes de segmentation décrites précédemment notamment leurs robustesses au bruit, nous les appliquons sur des images synthétiques, en fixant des paramètres qui contrôlent l'effet de la classification, le choix de ces derniers est d'une importance majeure, malheureusement, ces paramètres varient d'une image à une autre, ce qui rend, fastidieuse, la recherche de la valeur optimale de chaque paramètre. Pour ce travail, nous avons fixé les différents paramètres comme suit :

Le degré de flou  $m = 2$ .

Le critère d'arrêt  $\varepsilon = 0.001$ .

Les tests ont été implémentés sous Matlab R2013b, et exécutés sur un PC HP® Core (TM) i5-2400U CPU@1.60GHz 2.30 GHz).

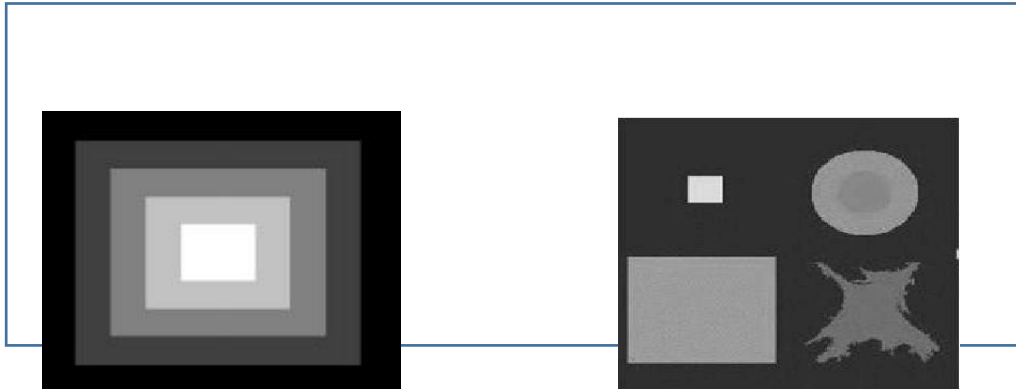


Figure II.4 : Images synthétiques avant la segmentation utilisées pour les tests Image synthétique 1 : composée de 5 régions ( $K = 5$ ) de forme carrée bien distinctes.

Image synthétique 2 : composée de 6 régions ( $K = 6$ ) de forme diverses (rectangle, carré, disque, anneau et une forme irrégulière) bien distinctes.

Durant notre simulation, nous avons pris en considération le critère d'optimisation de la qualité de segmentation pour la sélection des images segmentées. En effet, dans cette partie nous allons exposer les meilleurs résultats obtenus.



Figure II.5 : Résultats de la segmentation de l'image synthétique 1 par les deux méthodes



Figure II.6 : Résultats de la segmentation de l'image synthétique 2 par les deux méthode.

Les figures (II.6) et (II.7) représentent les techniques de segmentation définies précédemment qui aboutissent à des résultats pour toutes les images ayant des différentes classes tel que le résultat est un ensemble de clusters compacts et clairement séparés, sous réserve qu'on ait choisi la bonne valeur  $K$  du nombre de clusters.

### II.7.1 Choix de nombre $K$ de classes

Pour voir l'influence du nombre  $K$  de classes, nous appliquons les algorithmes décrits précédemment sur l'image synthétique 1 de la figure II.5. Nous faisons varier le nombre  $K$  de classes.

Image segmentée par les K-means pour  $K = 4$

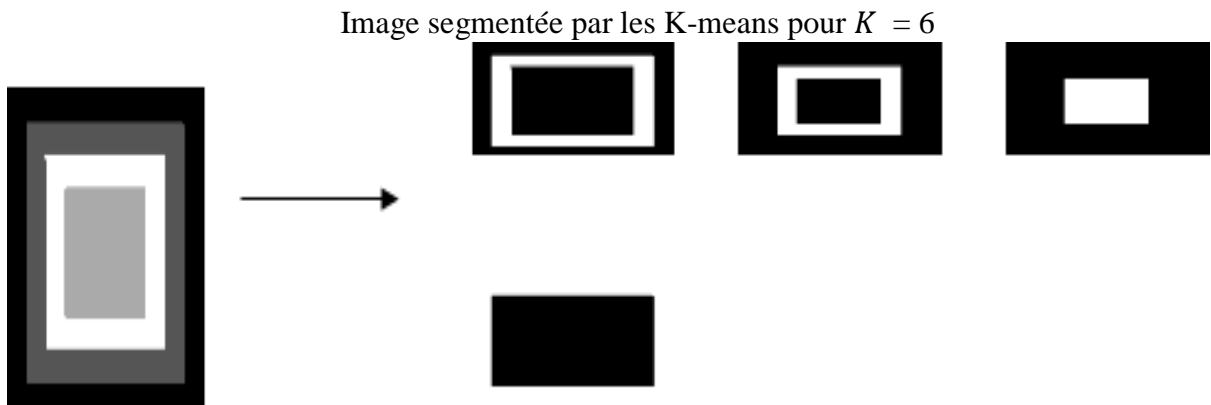
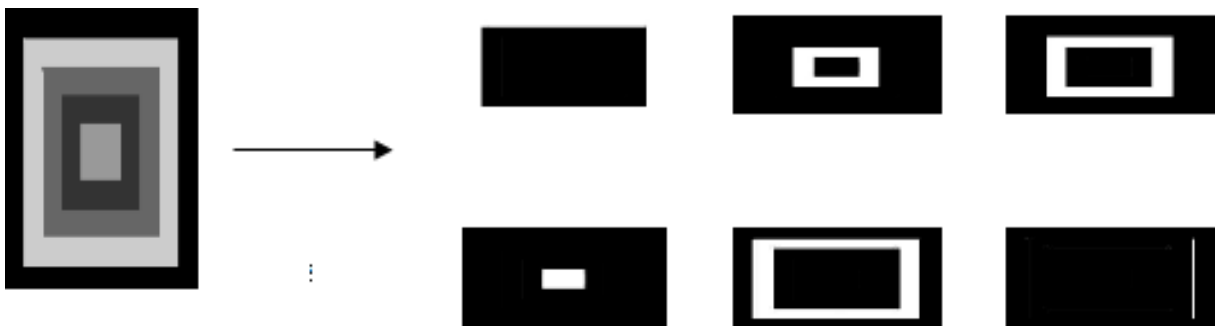


Figure II.7 : résultats de la segmentation par les K-means pour  $K = 4, K = 5, K = 6$



Image segmentée par les FCM pour  $K = 4$

Image segmentée par les FCM pour  $K = 5$ Image segmentée par les FCM pour  $K = 6$ *Figure II.8: résultats de la segmentation par les FCM pour  $K = 4$ ,  $K = 5$ ,  $K = 6$* 

Les figures (II.8) et (II.9) matérialisent les résultats de la segmentation pour différentes valeurs de  $K$  pour une image synthétique ayant cinq classes bien distinctes. Nous constatons bien que pour une valeur  $K = 5$ , les pixels de l'image sont classés selon 5 classes tel que l'image après la segmentation est semblable à l'image originale relativement aux formes et aux nombres de classe, Contrairement aux résultats obtenus lorsque  $k = 4$  et  $k = 6$ , nous obtenons à chaque fois une image non préservée en examinant les similarités de cette image avec l'image originale.

### II.7.2 Immunité au bruit

Pour tester l'immunité des algorithmes décrits précédemment vis-à-vis du bruit, nous avons ajouté à l'image synthétique 1 de la Figure II.5 un bruit blanc Gaussien de moyenne nulle et de variance relative à la valeur moyenne de l'image voir Figure II.10.

#### *Bruit Gaussien*

Le bruit Gaussien est un processus aléatoire qui suit une loi normale de moyenne et variance données, dans lequel la densité spectrale de puissance est la même pour toute les fréquences.

Les résultats observés sur la Figure II.9 montrent que l'image segmentée par la méthode des K-means est chevauchée, car cette méthode est très sensible au bruit, par contre en utilisant les FCM, nous observons bien une image peu bruitée.

## II.8. Conclusion

La segmentation d'images est au cœur de nombreux problèmes en imagerie médicale puisque bien souvent elle constitue la première étape d'un véritable flux de traitements d'image. Selon la suite à donner à ces images, certaines méthodes se révèlent plus probantes.

Le problème de la segmentation reste ouvert, au vu du grand nombre de publications régulièrement proposées sur le sujet. En effet, il n'existe pas de méthode générale fonctionnant pour tout type d'images, les techniques existantes étant optimisées pour une application donnée. Puisque la qualité du résultat est intrinsèquement subjective et que la finalité d'utilisation de la segmentation obtenue influe sur la perception de qualité de ce résultat, le choix d'une technique plutôt qu'une autre est assez ardu. Mais le choix d'une méthode dépend essentiellement des besoins de l'utilisateur.

Les techniques de classification ont pour principe, de regrouper les pixels suivant un critère de similarité prédéfini. Le fait la valeur  $K$  (nombre de classes) n'est pas bien choisie, le résultat de classification dépend fortement de ce paramètre, tel que sa bonne initialisation amènera probablement à une bonne classification. Du point de vue de l'immunité au bruit, nous constatons que le résultat de segmentation de l'algorithme FCM est bien meilleur comparé aux algorithmes K-Means.

---

---

*CHAPITRE III:*  
**LES RESEAUX DES NEURONES**

---

---

**CHAPITRE III****LES RESEAUX DES NEURONES****1 Les réseaux de neurones artificiels****INTRODUCTION :**

Un neurone est donc l'unité élémentaire de calcul d'un réseau de neurones. Son principe général est de retourner une information en sortie à partir de plusieurs informations en entrée. L'information entrante peut être, par exemple, issue de l'information sortante d'autres neurones dans le cadre d'un réseau. Plus précisément, nous notons  $x_1, \dots, x_n \in \mathbb{R}$  les informations entrantes et, pour chaque  $i \in \{1, \dots, n\}$ , nous associons un poids  $w_i \in \mathbb{R}$  à  $x_i$ . Contrairement aux poids que nous avons utilisés en cours, il faut noter que les  $w_i$  sont ici des nombres réels potentiellement négatifs et dont la somme ne vaut pas nécessairement 1. Nous introduisons également un poids  $w_0 \in \mathbb{R}$ , appelé coefficient de biais, associé à une information virtuelle  $x_0 = -1$  dont le rôle sera précisé ultérieurement. L'information traitée par le neurone n'est pas l'ensemble des  $x_i$  mais leur moyenne pondérée

**1. Définition :**

Les réseaux de neurones artificiels sont des réseaux fortement connectés de processeurs élémentaires fonctionnant en parallèle. Chaque processeur élémentaire calcule une sortie unique sur la base des informations qu'il reçoit.

Toute structure hiérarchique de réseaux est évidemment un réseau.

**2. Le modèle neurophysiologique :**

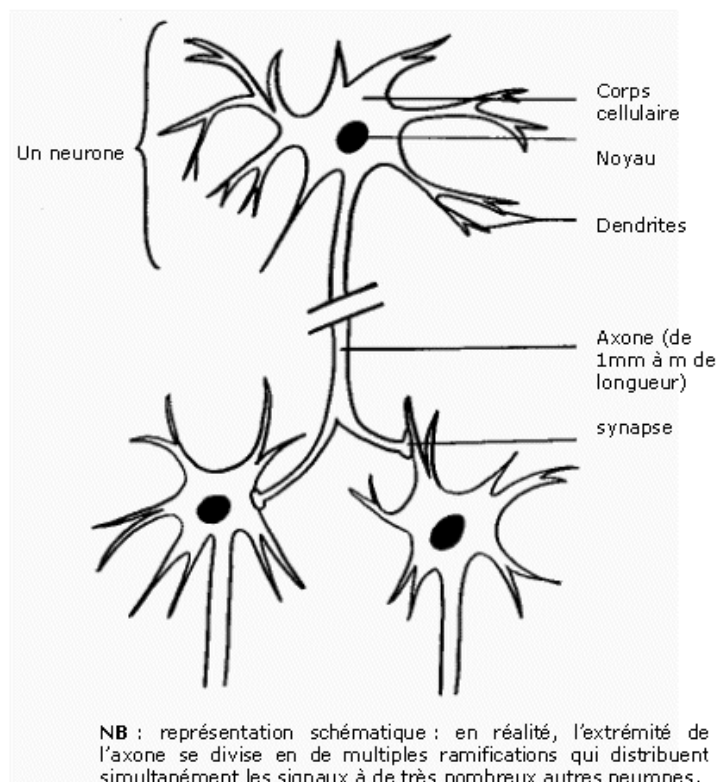
Le cerveau se compose d'environ 1012 neurones (mille milliards), avec 1000 à 10000 synapses (connexions) par neurone. Nous allons dans ce chapitre décrire succinctement l'élément de base du système nerveux central : le neurone. L'étape suivante nous conduit à l'étude de petits réseaux de neurones, tels ceux impliqués dans les arcs réflexes. Ceci nous amène à exposer les propriétés d'habituation, de sensibilisation et surtout à concevoir l'idée d'une modification physique des connexions entre neurones pour supporter ces phénomènes.

L'étude du mécanisme de la vision chez l'animal (et l'homme) permet d'appréhender les notions de messages somato-sensoriels, de réduction d'information, d'étages de traitement et de complexification de l'information

## 1 Le neurone :

### 1.1 Structure :

Le neurone est une cellule composée d'un corps cellulaire et d'un noyau. Le corps cellulaire se ramifie pour former ce que l'on nomme les dendrites. Celles-ci sont parfois si nombreuses que l'on parle alors de chevelure dendritique ou d'arborisation dendritique. C'est par les dendrites que l'information est acheminée de l'extérieur vers le soma, corps du neurone. L'information traitée par le neurone chemine ensuite le long de l'axone (unique) pour être transmise aux autres neurones. La transmission entre deux neurones n'est pas directe. En fait, il existe un espace intercellulaire de quelques dizaines d'Angstroms ( $10^{-9}$  m) entre l'axone du neurone afférent et les dendrites (on dit une dendrite) du neurone efférent. La jonction entre deux neurones est appelée la synapse



## 1.2 Physiologie

La physiologie du neurone est liée aux propriétés de la membrane nerveuse et au métabolisme de la cellule. La différence de potentiel mesurée entre le milieu intérieur de la cellule et le milieu extérieur est de  $-60$  mV. Pour maintenir une telle différence de potentiel, la cellule fait appel à des pompes ioniques ( $\text{Na}^+$ ,  $\text{K}^+$ , ...). Cependant, une faible dépolarisation de la membrane entraîne une certaine perméabilité aux ions sodiums ( $\text{Na}^+$ ), dont l'effet peut être catastrophique au niveau cellulaire.

En effet, à partir d'une certaine valeur seuil de dépolarisation de la membrane, il y a rupture des équilibres ioniques et création d'un potentiel d'action (aussi nommé "spike" en anglais).

### Modélisation mathématiques

Les réseaux de neurones biologiques réalisent facilement un certain nombre d'applications telles que la reconnaissance de formes, le traitement du signal, l'apprentissage par l'exemple, la mémorisation, la généralisation. Ces applications sont pourtant, malgré tous les efforts déployés en algorithmique et en intelligence artificielle, à la limite des possibilités actuelles. C'est à partir de l'hypothèse que le comportement intelligent émerge de la structure et du comportement des éléments de base du cerveau que les réseaux de neurones artificiels se sont développés.

Les réseaux de neurones artificiels sont des modèles, à ce titre ils peuvent être décrits par leurs Composants, leurs variables descriptives et les interactions des composants.

## 1 Composant (le neurone artificiel)

### 1.1 Structure

La figure 1 montre la structure d'un neurone artificiel. Chaque neurone artificiel est un processeur élémentaire. Il reçoit un nombre variable d'entrées en provenance de neurones amonts. A chacune de ces entrées est associée un poids ou abréviation de poids qui représente la force de la connexion. Chaque processeur élémentaire est doté d'une sortie unique, qui se ramifie ensuite pour alimenter un nombre variable de neurones avals. A chaque connexion est associée un poids.

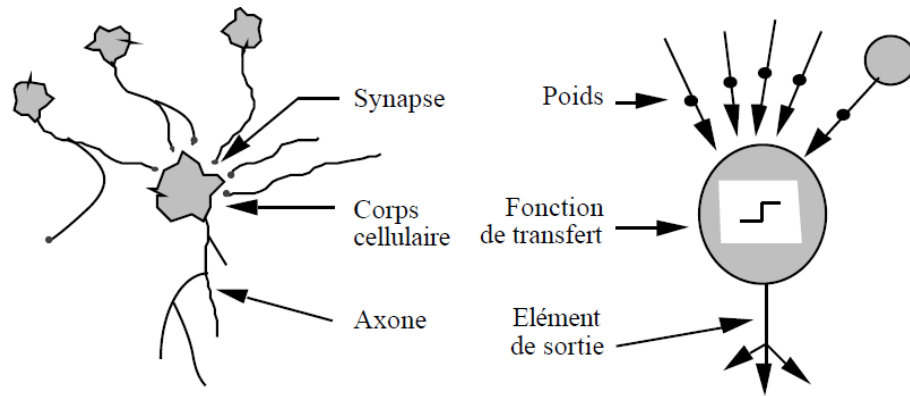


Figure III. 1 Neurone biologique / neurone artificiel

### 1.2 Comportement

On distingue deux phases. La première est habituellement le calcul de la somme pondérée des entrées ( $a$ ) selon l'expression suivante :  $a = \sum (w_i \cdot e_i)$

A partir de cette valeur, une fonction de transfert calcule la valeur de l'état du neurone. C'est cette valeur qui sera transmise aux neurones aval. Il existe de nombreuses formes possibles pour la fonction de transfert. Les plus courantes sont présentées sur la figure 3. On remarquera qu'à la différence des neurones biologiques dont l'état est binaire, la plupart des fonctions de transfert sont continues, offrant une infinité de valeurs possibles comprises dans l'intervalle  $[0, +1]$  (ou  $[-1, +1]$ ).

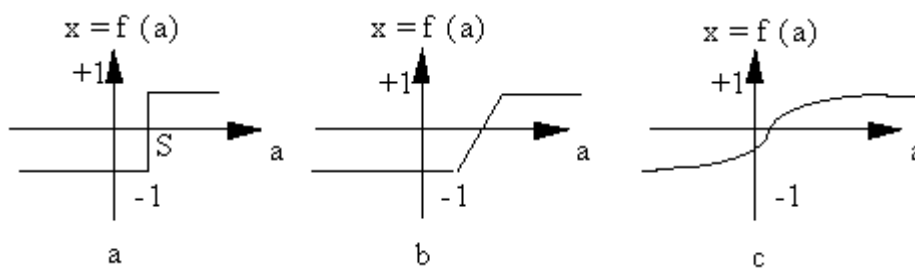


Figure III. 2 Différents types de fonctions de transfert pour le neurone artificiel, a : fonction à seuil ( $S$ , la valeur du seuil), b : linéaire par morceaux, c : sigmoïde

On constate que les équations décrivant le comportement des neurones artificiels n'introduisent pas la notion de temps. En effet, et c'est le cas pour la plupart des modèles

Actuels de réseaux de neurones, nous avons affaire à des modèles à temps discret, synchrone, dont le comportement des composants ne varie pas dans le temps.

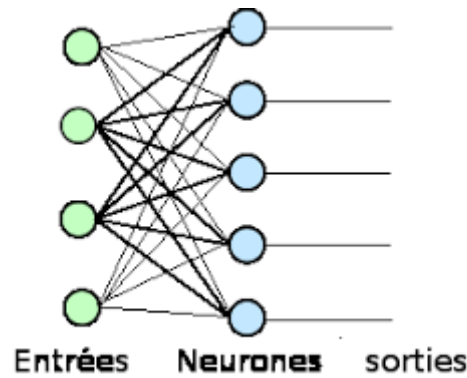


Figure III. 3 Perceptron mono-couche à  $n = 4$  entrées et  $p = 5$  sorties

## 2 Variables descriptives :

Ces variables décrivent l'état du système. Dans le cas des réseaux de neurones qui sont des systèmes non autonomes, un sous-ensemble des variables descriptives est constitué par les variables d'entrée, variables dont la valeur est déterminée extérieurement au modèle.

## 3 Fonctionnement

### 3.1 Perceptron

Le premier réseau de neurones que nous allons voir est le perceptron mono-couche. Les neurones ne sont pas, à proprement parlé, en réseau mais ils sont considérés comme un ensemble, nous considérons  $n$  variables d'entrée  $x_1, \dots, x_n \in \mathbb{R}$ . Le perceptron mono-couche est composé de  $p$  neurones, chacun étant connecté à toutes les variables d'entrée. Globalement, ce réseau dispose donc de  $n$  entrées et de  $p$  sorties et il peut être représenté comme sur la Figure 3.

Pour les notations, nous écrirons  $x = (x_1, \dots, x_n) \in \mathbb{R}^n$  pour le vecteur des entrées,  $y = (y_1, \dots, y_p) \in [0, 1]^p$  pour celui des sorties et, pour tout  $j \in \{1, \dots, p\}$ ,  $w_j \in \mathbb{R}$  est le coefficient de biais du  $j$ ème neurone. De plus, pour tout  $i \in \{1, \dots, n\}$  et tout  $j \in \{1, \dots, p\}$ ,  $w_{ji} \in \mathbb{R}$  est le poids associé à  $x_i$  par le  $j$ ème neurone. Nous pouvons ainsi définir la matrice  $W = (w_{ji})_{i,j}$  des poids de taille  $n \times p$ .

### 3.2 Apprentissage supervisé :

Le réseau de neurones du perceptron mono-couche permet de construire une procédure de classification en  $p$  classes en considérant chaque neurone comme un "indicateur" d'une

classe. La classe affectée à une série d'entrées est celle associée au neurone retournant la sortie maximale. Construire une telle procédure revient à choisir des coefficients de biais  $w_0, \dots, w_p \in \mathbb{R}$  et une matrice  $W$  des poids. Comme pour les autres procédures d'apprentissage supervisé, nous allons construire celle-ci à partir d'un jeu de données étiquetées en considérant des poids et des coefficients de biais qui rendent un certain critère d'erreur (moindres carrés, entropie, ...) minimal. Nous ne développerons pas ici les algorithmes d'optimisation utilisés en pratique pour trouver  $W$  et les  $w_j$ . Les approches les courantes sont basées sur des méthodes de descente de gradient

#### 4 Perceptron multicouche :

Il est possible de généraliser le perceptron en empilant plusieurs perceptrons mono-couches. De cette façon, les sorties d'une couche sont les entrées de la suivante. Ce réseau de neurone est, bien entendu, plus compliqué mais s'utilise de manière similaire à un perceptron mono-couche. L'intérêt principal est d'être capable d'approcher des comportements moins linéaires et d'obtenir ainsi des erreurs plus faibles sur les données d'entraînement (forte adéquation aux données) au prix d'une complexité plus grande (voir dernier chapitre du cours). Dans le cas d'un perceptron à deux couches ayant  $n$  entrées et  $p$  sorties, il nous faut donc choisir le nombre de neurones à mettre dans la couche intermédiaire. En pratique, il vaut toujours mieux avoir trop de neurones cachés que pas assez si nous souhaitons capter les phénomènes non-linéaires. Cependant, si nous utilisons un trop grand nombre de neurones intermédiaires, nous risquons le phénomène d'over fitting sur nos données d'apprentissage. Il est possible d'utiliser la validation croisée pour trouver une solution à ce problème pratique. Le paramètre de neuralnet qui permet d'indiquer le nombre de neurones cachés est `hidden`. Pour construire le réseau à deux couches avec 2 neurones cachés, entrez la commande suivante, `multi <- neuralnet (infert_form, data = infert, hidden = 2)`.

#### Exemple théorique

Soit un réseau multicouche composé de 361 ( $19 \times 19$ ), 25 et 361 neurones. Ce réseau a appris à associer à la lettre "a" présentée en entrée la même lettre en sortie (fig.4). Présentons au Réseau cette lettre avec quelques erreurs : un certain nombre de pixels ont été inversé (ils sont passés de blanc à noir ou inversement). L'image est composée de  $19 \times 19$  pixels, chacun de ces pixels est associé à un neurone de la couche d'entrée. Chacun des 25 neurones de la couche cachée reçoit 361 connexions (une pour chaque neurone d'entrée) et envoie sa sortie à chacun des neurones de la couche de sortie (au nombre de 361). Dans notre exemple, la couche

cachée se compose de 25 neurones, mais ce nombre, à la différence des couches d'entrée et de sortie, n'est pas impératif il y a donc  $(361 \cdot 25) = 18050$  connexions dans le réseau.

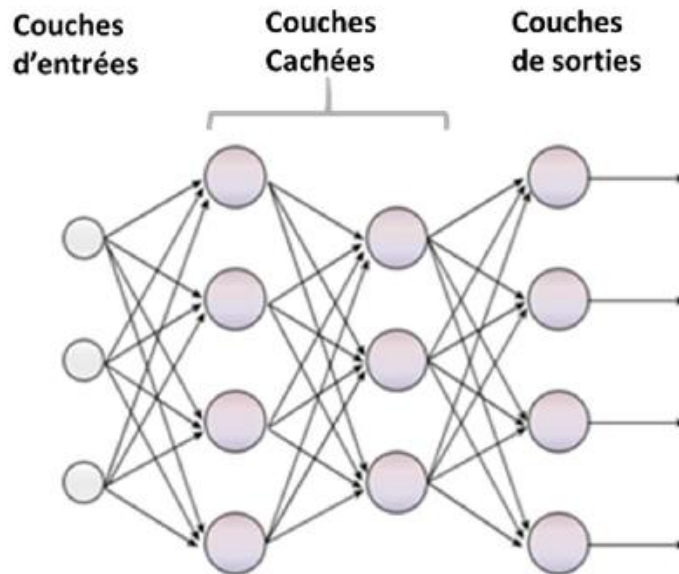


Figure III.4 Comportement en phase de reconnaissance d'un réseau de neurone multicouche

Lors d'une tâche d'auto-association. Les neurones sont binaires. La valeur d'activation de chaque Neurone est indiquée par la hauteur de la colonne. Les neurones sont rangés par couche, tous Les neurones d'une couche sont connectés à tous les neurones de la couche suivante (avale).

La première étape code l'image d'entrée sur le réseau. Il s'agit pour chaque neurone de la couche d'entrée de fixer la valeur de son état selon la couleur du pixel correspondant. Si les neurones qui composent le réseau sont binaires, on choisit arbitrairement de coder un pixel noir par un niveau d'activation du neurone égal à 1 ; si le pixel est blanc alors le niveau d'activation du neurone correspondant est égal à 0.

La seconde étape est celle du calcul de la réponse du réseau qui se décompose en autant de sous-étapes qu'il y a de couches dans le réseau. Ainsi, chaque neurone de la couche d'entrée envoie sa valeur aux neurones de la couche cachée. Chacun des neurones de la couche cachée est en fait un Perceptron à 361 entrées. Chacun des neurones réalise la somme pondérée de ses Entrées et seille. Ce processus est effectué en parallèle et indépendamment pour tous les Neurones de la couche cachée. Lorsque le vecteur d'activation de la couche cachée a été obtenu, Le même processus est répété avec les neurones de la couche de sortie. On considère ceux-ci Comme 361 Perceptrons indépendants à 25 entrées.

La dernière étape est l'interprétation du vecteur d'activation de la couche de sortie par l'expérimentateur. Dans notre cas, on réalise l'opération inverse du codage initial, à savoir Associer un pixel noir à chacun des neurones de la couche de sortie dont la valeur d'activation est égale à 1, un pixel blanc dans le cas contraire.

La première étape code l'image d'entrée sur le réseau. Il s'agit pour chaque neurone de la couche d'entrée de fixer la valeur de son état selon la couleur du pixel correspondant. Si les neurones qui composent le réseau sont binaires, on choisit arbitrairement de coder un pixel noir par un niveau d'activation du neurone égal à 1 ; si le pixel est blanc alors le niveau d'activation du neurone correspondant est égal à 0.

La seconde étape est celle du calcul de la réponse du réseau qui se décompose en autant de Sous-étapes qu'il y a de couches dans le réseau. Ainsi, chaque neurone de la couche d'entrée envoie sa valeur aux neurones de la couche cachée. Chacun des neurones de la couche cachée est en fait un Perceptron à 361 entrées. Chacun des neurones réalise la somme pondérée de ses entrées et seuil. Ce processus est effectué en parallèle et indépendamment pour tous les neurones de la couche cachée. Lorsque le vecteur d'activation de la couche cachée a été obtenu, Le même processus est répété avec les neurones de la couche de sortie. On considère ceux-ci Comme 361 Perceptrons indépendants à 25 entrées.

La dernière étape est l'interprétation du vecteur d'activation de la couche de sortie par L'expérimentateur. Dans notre cas, on réalise l'opération inverse du codage initial, à savoir. Associer un pixel noir à chacun des neurones de la couche de sortie dont la valeur d'activation est égal à 1, un pixel blanc dans le cas contraire.

## 6 Comportements combinatoire et séquentiel (TD)

Parmi les variables descriptives qui ne sont pas des variables d'entrée, on appelle variables d'état les variables dont la valeur à n'importe quelle date, ajoutée à la valeur des entrées, déterminent de manière unique les valeurs de toutes les autres. Les états caractérisent les possibilités de mémorisation du système : l'information qu'il peut retenir des stimuli passés et qui modifiera la réponse à des stimuli futurs. Un système est sans mémoire s'il n'y a pas de variables d'état. Pour un tel système, les réponses présentes et futures ne peuvent en aucune manière être affectées par des stimuli passés. De tels systèmes sont appelés combinatoires, car leur réponse à n'importe quelle date est uniquement fonction du stimulus reçu à cet instant.

Le comportement de ces systèmes est défini par l'équation (F est la fonction réalisée,

$E(t)$  est l'entrée,  $S(t)$  est la sortie) :  $S(t) = F(E(t))$

Un grand nombre de modèles neuronaux, parmi les plus utilisés, n'ont pas de variables d'état et montrent donc un comportement combinatoire : réseau multicouche, carte auto organisatrice, réseau ART1, ...

Inversement, un système à états répondra différemment à des entrées présentes et identiques selon l'histoire de ses stimuli d'entrées passés. Son comportement, nommé comportement séquentiel, est donc défini par une équation de la forme  $S(t) = F(E(t), E(t-1), E(t-2), \dots, E(1), E(0))$

#### **4 Apprentissage :**

L'apprentissage est une phase du développement d'un réseau de neurones durant laquelle le Comportement du réseau est modifié jusqu'à l'obtention du comportement désiré.

L'apprentissage neuronal fait appel à des exemples de comportement.

Dans le cas des réseaux de neurones artificiels, on ajoute souvent à la description du modèle l'algorithme d'apprentissage. Le modèle sans apprentissage présente en effet peu d'intérêt. Dans la majorité des algorithmes actuels, les variables modifiées pendant l'apprentissage sont les poids des connexions. L'apprentissage est la modification des poids du réseau dans l'optique d'accorder la réponse du réseau aux exemples et à l'expérience. Il est souvent Impossible de décider à priori des valeurs des poids des connexions d'un réseau pour une application donnée. A l'issue de l'apprentissage, les poids sont fixés : c'est alors la phase d'utilisation. Certains modèles de réseaux sont improprement dénommés à apprentissage permanent. Dans ce cas il est vrai que l'apprentissage ne s'arrête jamais, cependant on peut toujours distinguer une phase d'apprentissage (en fait de remise à jour du comportement) et une phase d'utilisation. Cette technique permet de conserver au réseau un comportement adapté malgré les fluctuations dans les données d'entrées Au niveau des algorithmes d'apprentissage, il a été défini deux grandes classes selon que l'apprentissage est dit supervisé ou non supervisé. Cette distinction repose sur la forme des exemples d'apprentissage. Dans le cas de l'apprentissage supervisé, les exemples sont des couples (Entrée, Sortie associée) alors que l'on ne dispose que des valeurs (Entrée) pour l'apprentissage non supervisé. Remarquons cependant que les modèles à apprentissage non la loi de Hebb, un exemple d'apprentissage non supervisé.

Elle s'exprime de la façon suivante : "Si 2 cellules sont activées en même temps alors la force de la connexion augmente» . La modification de poids dépend de la coactivation des neurones présynoptique et post synaptique, ainsi que le montre la table 1.  $x_i$  et  $x_j$  sont respectivement les valeurs d'activation Des neurones  $i$  et  $j$ ,  $\Delta w_{ij}$  (dérivée partielle du poids) correspond à la modification de poids réalisée.

| $x_i$ | $x_j$ | $\Delta w_{ij}$ |
|-------|-------|-----------------|
| 0     | 0     | 0               |
| 0     | 1     | 0               |
| 1     | 0     | 0               |
| 1     | 1     | +               |

Tableau 3.1 Loi de hebb

La loi de Hebb peut être modélisée par les équations suivantes ( $w(t+1)$  est le nouveau poids,  $w(t)$  l'ancien) :

$$w_{ij}(t+1) = w_{ij}(t) + \Delta w_{ij}(t)$$

$$w_{ij}(t) = x_i \cdot x_j \text{ (la captivité est modélisée comme le produit des deux valeurs d'activation)}$$

L'algorithme d'apprentissage modifie de façon itérative (petit à petit) les poids pour adapter la réponse obtenue à la réponse désirée. Il s'agit en fait de modifier les poids lorsqu'il y a erreur seulement

1/ Initialisation des poids et du seuil  $S$  à des valeurs (petites) choisies au hasard.

2/ Présentation d'une entrée  $E_i = (e_1, \dots, e_n)$  de la base d'apprentissage.

3/ Calcul de la sortie obtenue  $x$  pour cette entrée :

$$a = \sum (w_i \cdot e_i) - S \text{ (la valeur de seuil est introduite ici dans le calcul de la somme}$$

Pondérée)

$x = \text{signe}(a)$  ( si  $a > 0$  alors  $x = +1$  sinon  $a \leq 0$  alors  $x = -1$  )

4/ Si la sortie  $x$  est différente de la sortie désirée  $d_i$  pour cet exemple d'entrée  $E_i$  alors modification des poids ( $\mu$  est une constante positive, qui spécifie le pas de modification des

Poids) :

$$w_{ij}(t+1) = w_{ij}(t) + \mu \cdot (x_i \cdot x_j) .$$

Choisissons pour les neurones un comportement binaire. Les entrées  $e_1$  et  $e_2$  sont considérées comme des neurones

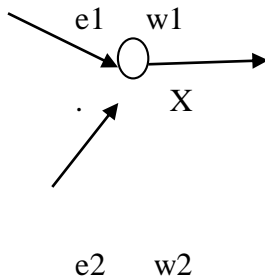


Figure 2. Réseau de 3 neurones (les 2 entrées sont considérées comme deux neurones)

Pour la résolution du problème on va déterminer la table n2. Nous allons réaliser l'apprentissage sur un problème très simple. La base d'apprentissage est décrite par la table 2 :

| e1 | e 2 | x     |
|----|-----|-------|
| 1  | 1   | 1(1)  |
| 1  | -1  | 1(2)  |
| -1 | 1   | -1(3) |
| -1 | -1  | -1(4) |

Tableau 3.2. Base d'exemples d'apprentissage pour la loi de Hebb.

1/ Conditions initiales :  $\mu = +1$ , les poids et le seuil sont nuls.

2/ on Calcule la valeur de x pour l'exemple (1) :

$$3/ a = w1.e1 + w2.e2 - S = 0.0 \cdot 1 + 0.0 \cdot 1 - 0.0 = 0 \text{ a } \neq 0 \Rightarrow x = -1$$

4/ La sortie est fausse, il faut donc modifier les poids en appliquant :

$$w1 = w1 + e1.x = 0.0 + 1.1 = 1$$

$$w2 = w2 + e2.x = 0.0 + 1.1 = 1.$$

## 5 Mémoires associatives :

### Fonctionnement

Le principe de fonctionnement d'une mémoire associative se résume ainsi. Soit (E1, E2, ..., E1, ...) un ensemble de vecteurs de Rn. A chaque vecteur E1 appelé "prototype" de l'espace d'entrée est associé un vecteur de sortie S1. La relation d'association entre E1 et S1 est linéaire. Elle est donnée par l'équation :  $S1 = W \cdot E1$  Où W est la matrice des poids de dimension (p.n). C'est une matrice rectangulaire de p lignes et n colonnes.

L'objectif est de faire réaliser à ce réseau des associations entre les vecteurs d'entrées et les Vecteurs de sortie désirés. Ceci nécessite une étape d'apprentissage.

### Conclusion :

L'objectif pédagogique visé dans ce survol du monde biologique est la mise en exergue d'une organisation structurelle des neurones. Chaque structure est dotée d'une fonction Particulière et ces structures adaptent leur comportement par des mécanismes d'apprentissage.

L'apprentissage implique des modifications physiques des connexions entre neurones. L'association entre plusieurs structures neuronales, dotées chacune d'une fonction précise, Permet l'émergence d'une fonction d'ordre supérieure pour l'ensemble.

---

---

# *CHAPITRE IV:*

**APPLICATION D'UNE APPROCHE NEURONALE  
POUR LA SEGMENTATION DES IMAGES  
MEDICALES**

---

---

## CHAPITRE 4

# APPLICATION D'UNE APPROCHE NEURONALE POUR LA SEGMENTATION DES IMAGES MEDICALES

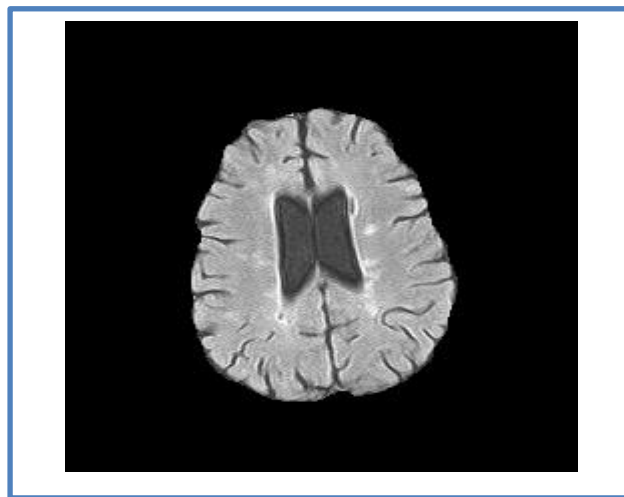
### IV.1. Introduction

Dans ce chapitre nous allons exposer les résultats obtenus par les différentes techniques de segmentations des images médicales. Ce chapitre a objectif de présenter la classification des images médicales par les réseaux de neurones RBF afin de rendre cette dernière plus efficace pour la classification.

En premier temps, nous allons appliquer les techniques : FCM (Fuzzy C-Means) et K-means, en deuxième temps, on va appliquer une technique intelligente basée sur les réseaux de neurones RBF. Finalement, on va présenter une étude comparative entre les différentes techniques présentées.

### IV.2. Techniques proposées

Les images prises pour le test dans ce chapitre présentent un IRM cérébrales, on applique les différentes techniques sur ces images sans bruit en premier temps, et avec bruit en deuxième temps.



*Figure 4.1 : Image IRM cérébrale originale sans bruit*

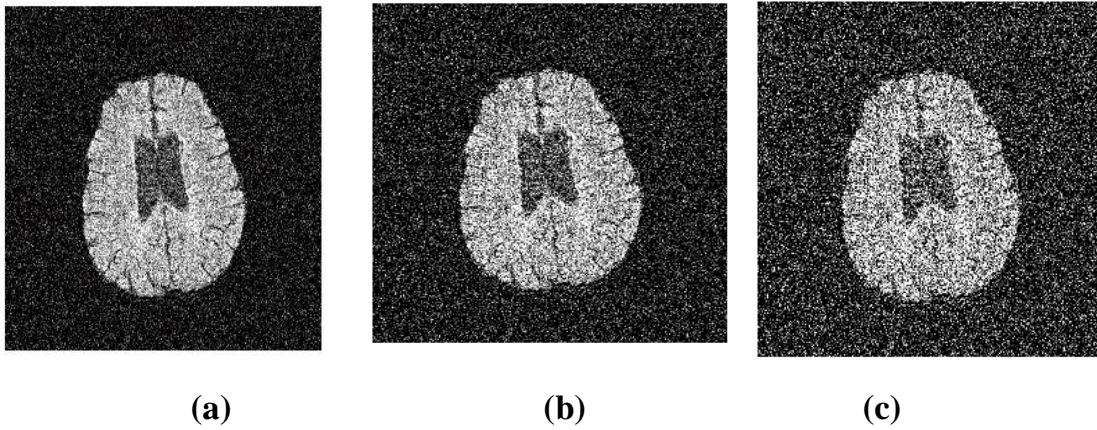


Figure 4.2 : Image IRM cérébrale originale avec bruit gaussien (a :5%, b :10%, 20%)

### IV.2.1. Algorithme K-means

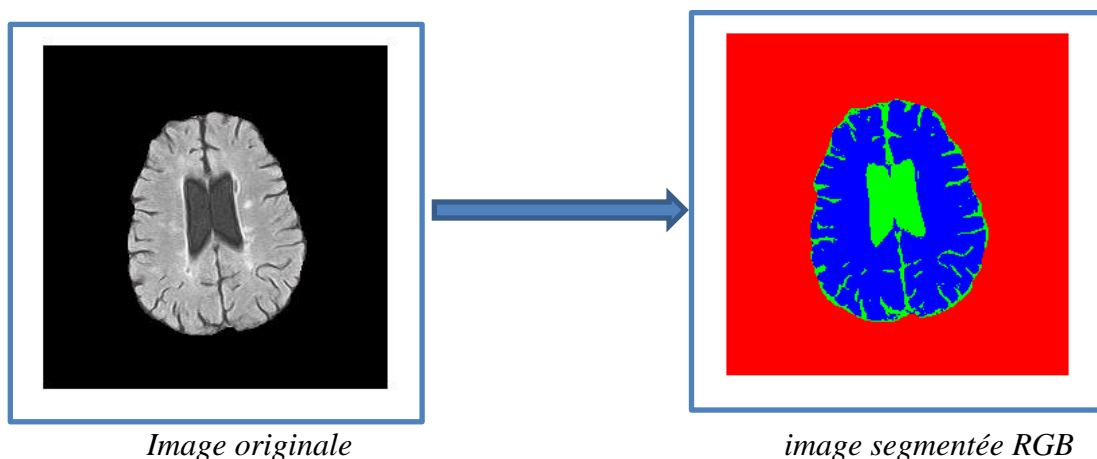
Les principales étapes de l'algorithme k-means sont :

- Choix aléatoire de la position initiale des K clusters.
- Affecter les objets à un cluster suivant un critère de minimisation des distances (Généralement selon une mesure de distance euclidienne).
- Une fois tous les objets placés, recalculer les K centroïdes.
- Répéter les étapes 2 et 3 de l'algorithme présenté dans le chapitre 2 jusqu'à ce que plus aucune réaffectation ne soit faite

#### Résultats d'extraction des classes :

Nous avons utilisé MATLAB pour réaliser ce travail, qui est un environnement interactif permettant d'effectuer des calculs numériques complexes particulièrement très utiles dans le domaine de Traitement d'images.

#### A. Sans bruit :



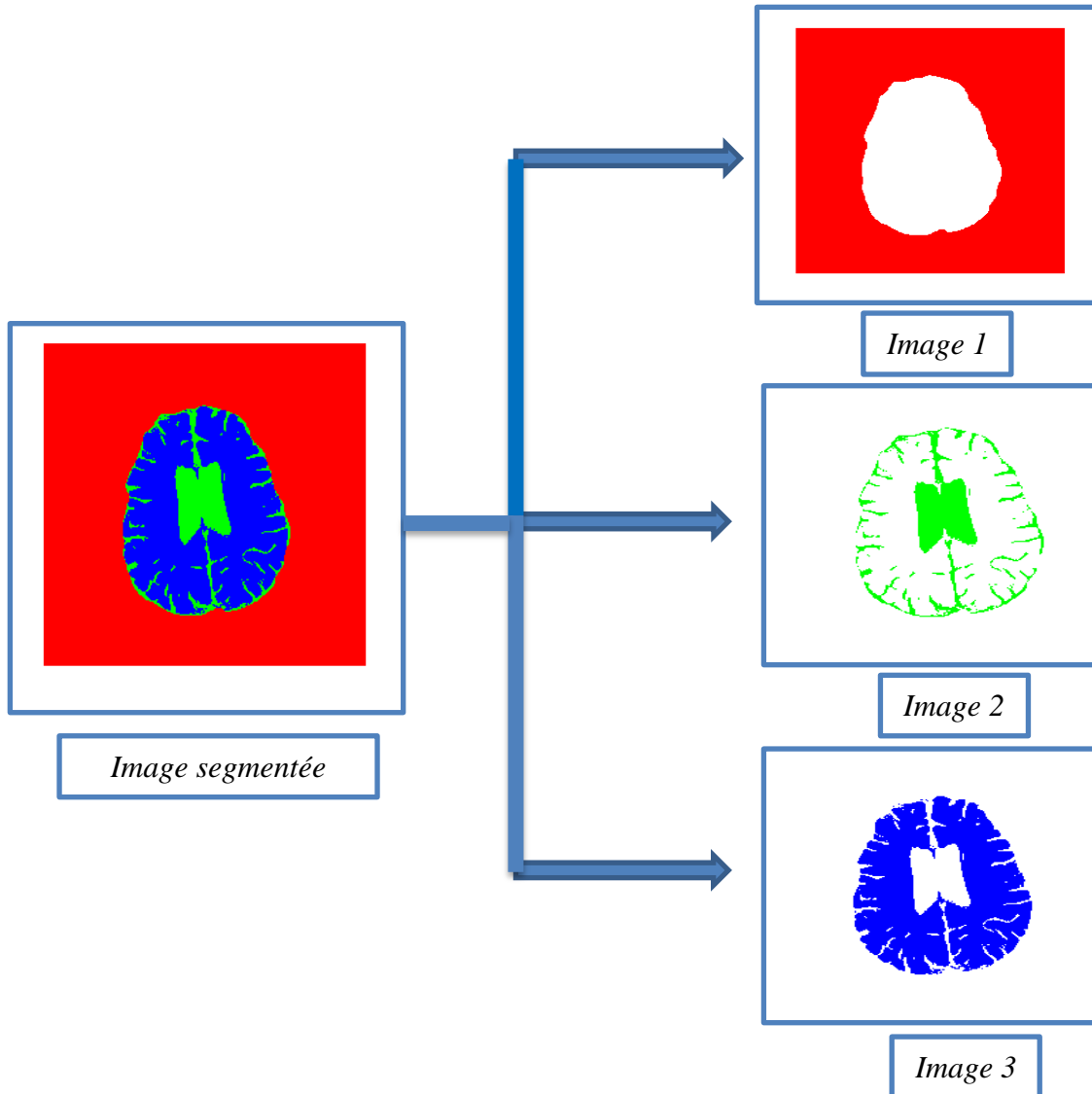


Figure 4.3 : Résultats de segmentation obtenus par l'approche K-means en absence du bruit

Après la classification de l'image IRM cérébrale originale, on compare les différentes classes avec une vérité terrain (classes de référence), d'abord on va convertir les images classifiés RGB en images binaires.

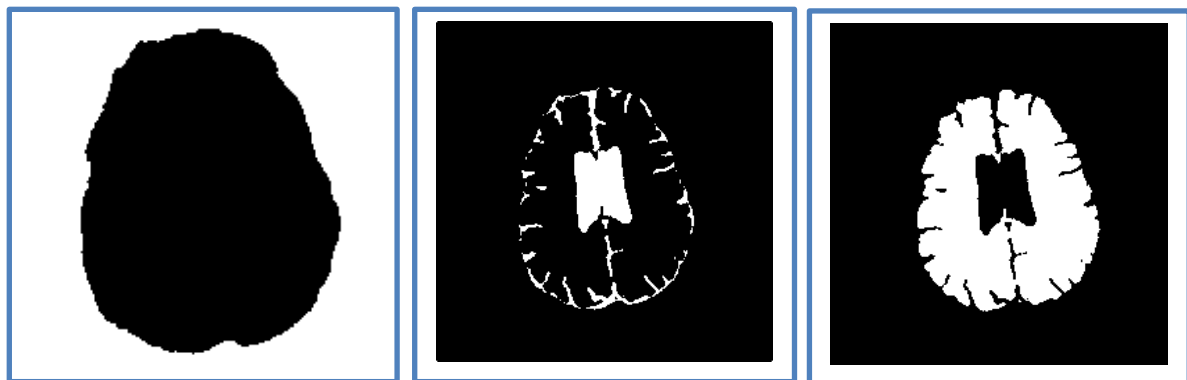


Figure 4.4 : Classes de l'image IRM cérébrale originales



Figure 4.5: Classes de l'image IRM par l'approche K-means sans bruit

**B. Avec bruit gaussien (5%):**

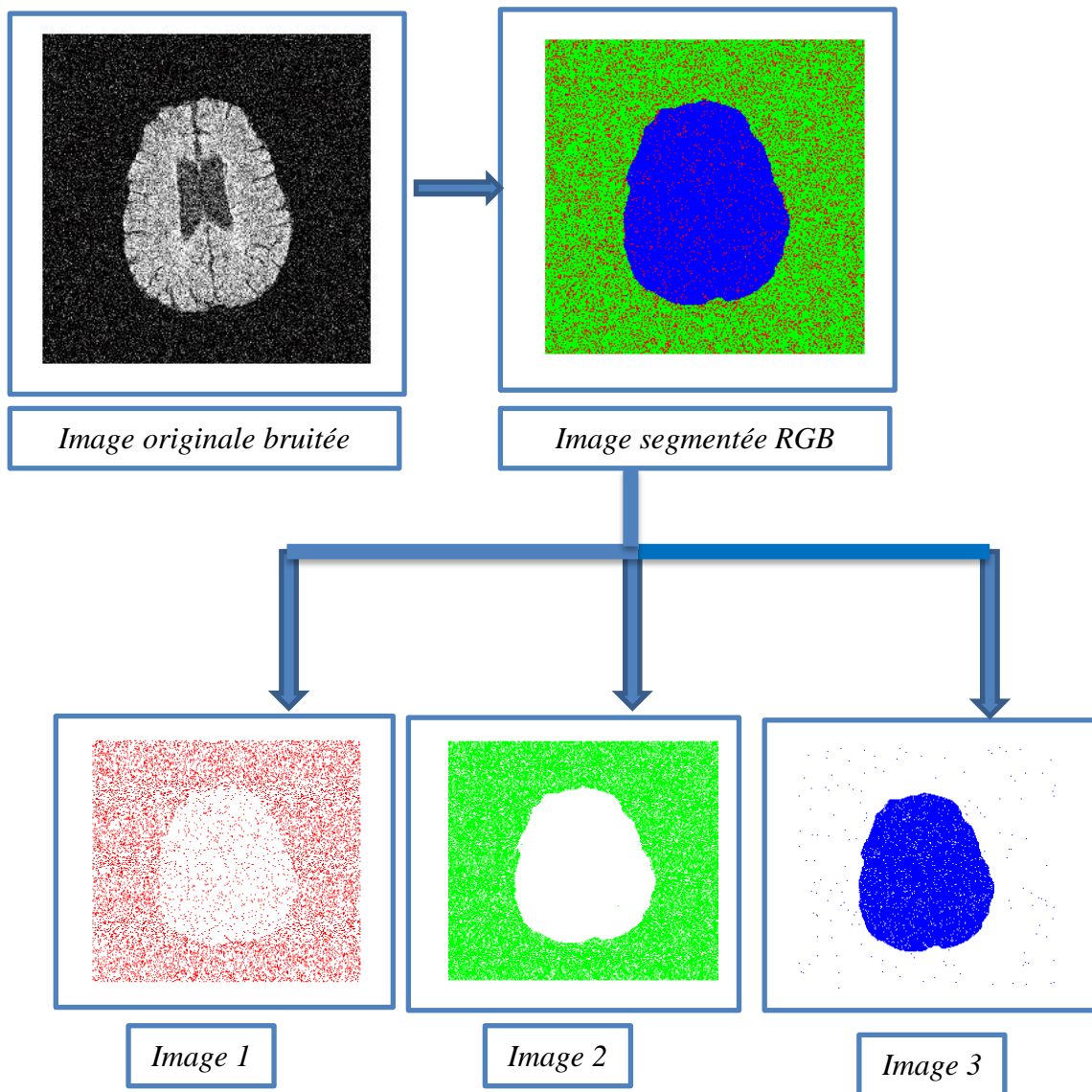


Figure 4.6: Résultats de segmentation par l'approche K-means en présence du bruit gaussien 5%

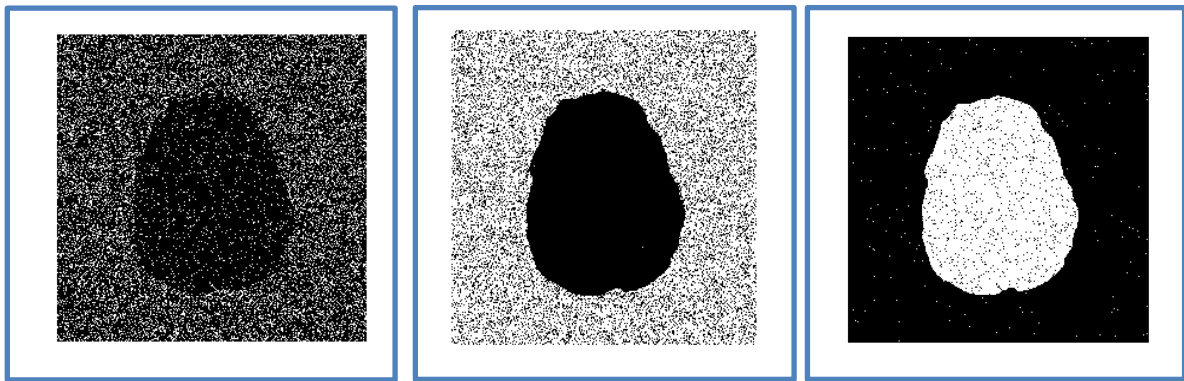


Figure 4.7: Classes de l'image IRM par l'approche K-means avec bruit gaussien 5% (images binaires)

C. Avec bruit gaussien (10%):

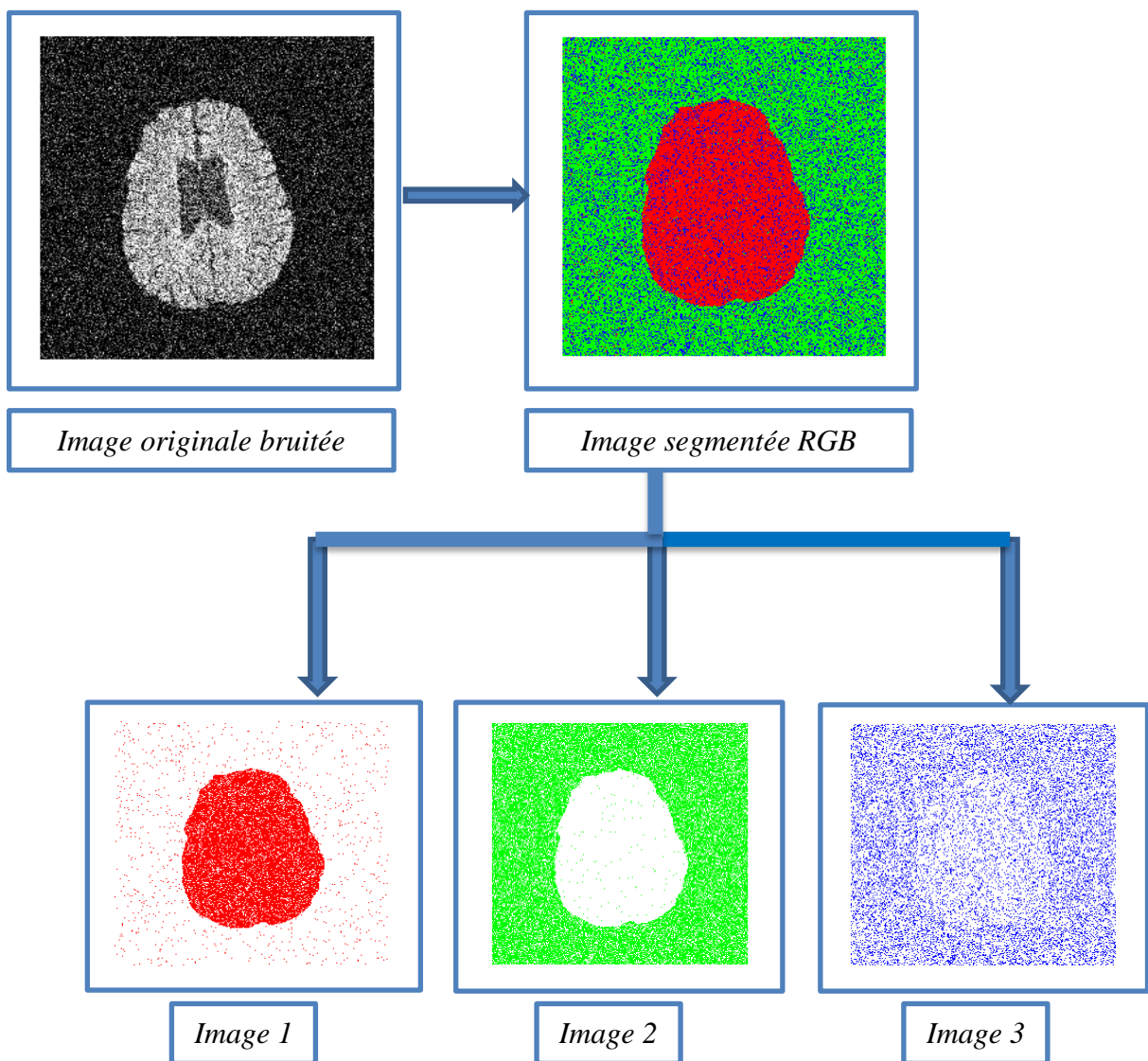


Figure 4.8: Résultats de segmentation par l'approche K-means en présence du bruit gaussien 10%

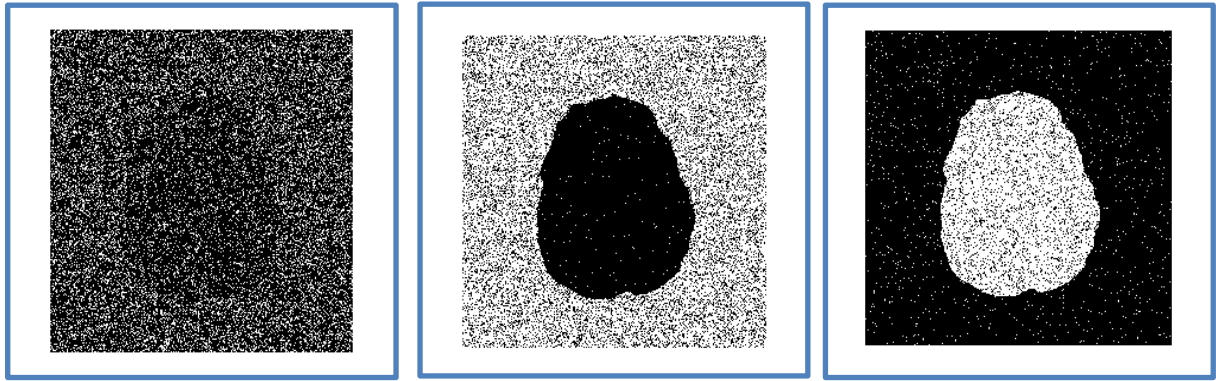


Figure 4.9: Classes de l'image IRM par l'approche K-means avec bruit gaussien 10% (images binaires)

Les résultats de la segmentation obtenus dans la figure (4.5) visuellement, nous pouvons voir que K-means donne des meilleurs résultats quand on les compare avec les images originales données dans la figure (4.4). Dans le cas où l'image est affectée par un bruit gaussien les performances de l'approche diminuent mais avec des contours acceptables, alors cette approche perd ces performances de classification en présence de bruit.

#### IV.2.2. Algorithme FCM (Fuzzy C-Means)

Nous utilisons l'algorithme FCM, décrit dans le chapitre (2) pour segmenter les images cérébrales. Pour cela, nous présentons les différents paramètres liés à l'algorithme à savoir :

- **Etape 1** : fixer les paramètres  $C$  : nombre de classe  $\varepsilon$  : seuil représentant l'erreur de convergence (par exemple  $\varepsilon = 0.001$ ),  $m$  : degré de flou, généralement  $m = 2$ .
- **Etape 2** : Initialiser la matrice de degrés d'appartenances  $U$  par des valeurs aléatoires dans l'intervalle  $[0, 1]$ .
- **Etape 3** : calculer le prototype  $v$  par la relation (2.7). Calculer  $(u(\cdot), v)$  par la relation (2.5).
- **Etape 4** : mettre à jour la matrice de degrés d'appartenance par la relation (2.6). Calculer  $\bar{J}_m(u(\cdot), v)$  par la relation (2.5).
- **Etape 5** : répéter les étapes 3 à 4 jusqu'à satisfaction du critère d'arrêt

#### Résultats d'extraction des classes :

##### A. Sans bruit :

En premier temps, on prend que l'image n'est pas affecté par un bruit.

Les paramètres de cette approche sont fixés comme suit :  $C=3$ ,  $\varepsilon=0.00001$ ,  $m=2$ .

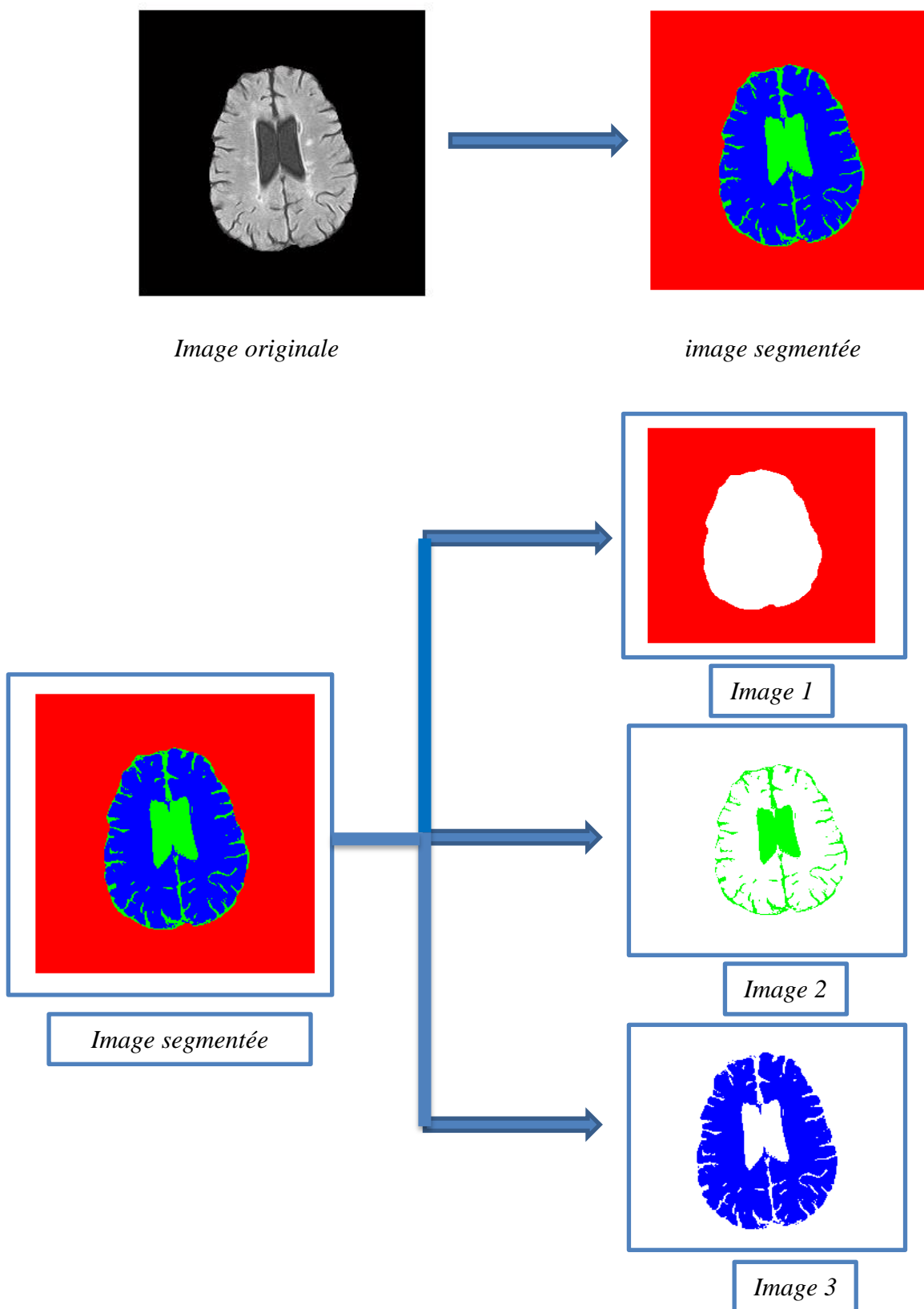


Figure 4.10 : Résultats de segmentation obtenus par l'approche FCM en absence du bruit



Figure 4.11: Classes de l'image IRM par l'approche FCM sans bruit

A. Avec bruit gaussien (5%):

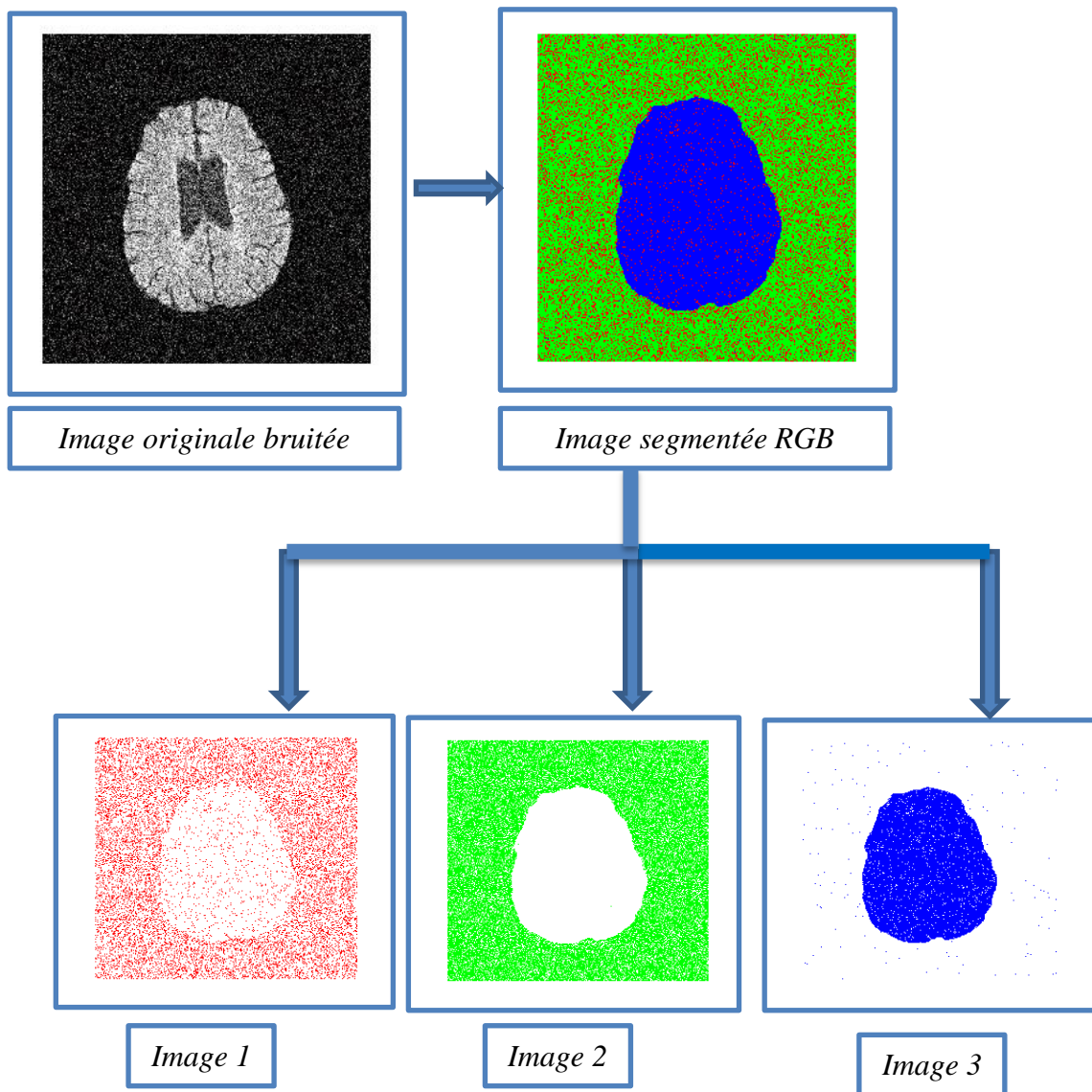


Figure 4.12: Résultats de segmentation par l'approche FCM en présence du bruit gaussien 5%

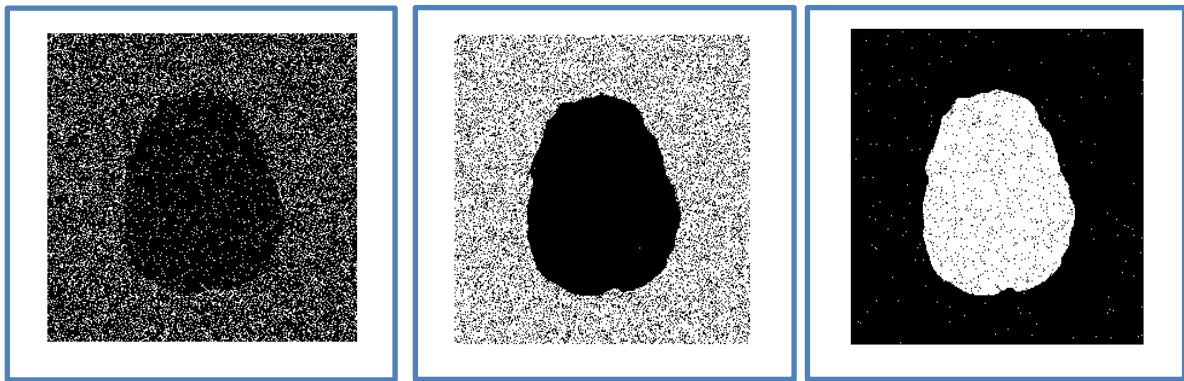


Figure 4.13: Classes de l'image IRM par l'approche FCM avec bruit gaussien 5% (images binaires)

**B. Avec bruit gaussien (10%):**

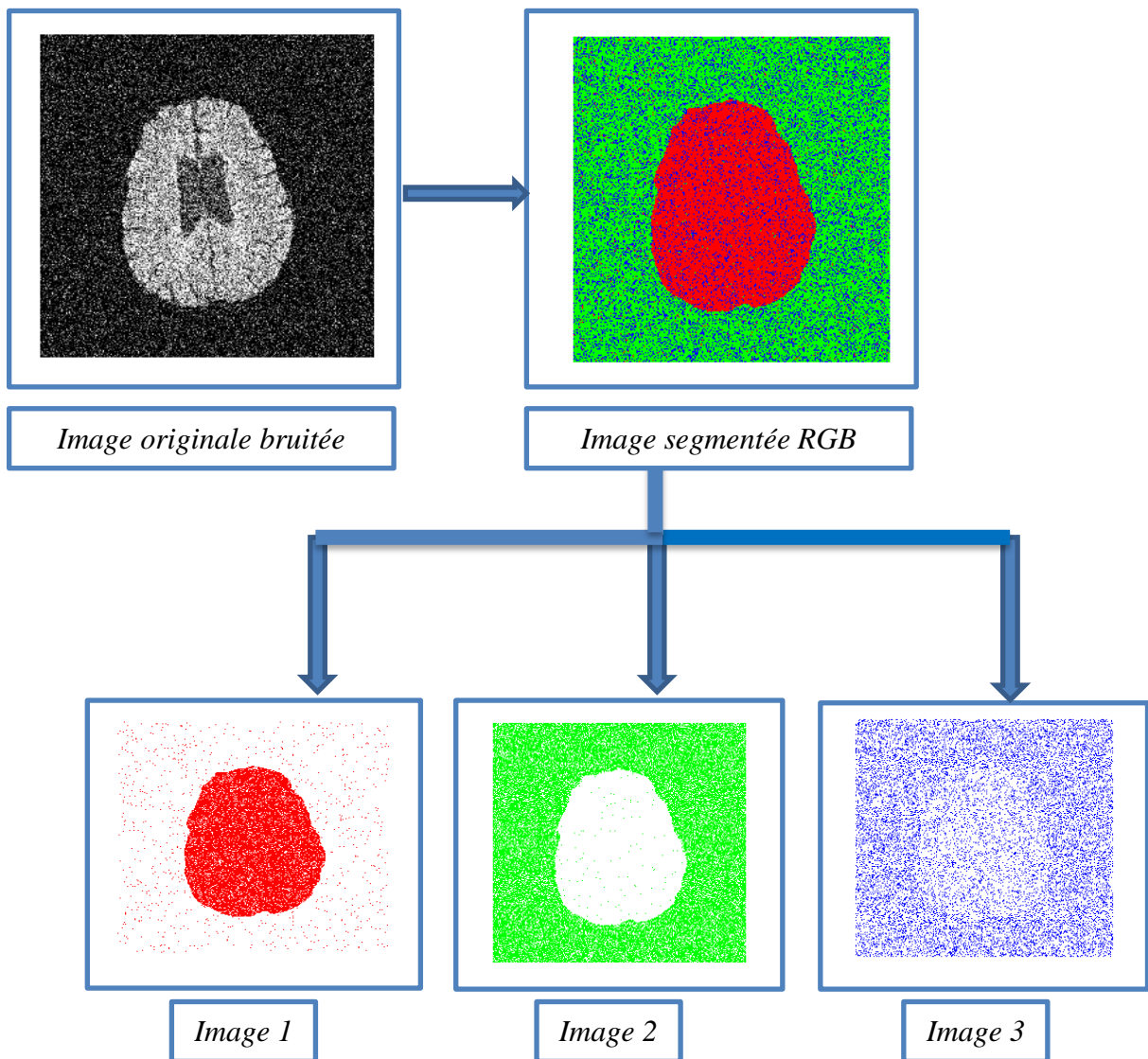


Figure 4.14: Résultats de segmentation par l'approche FCM en présence du bruit gaussien 10%

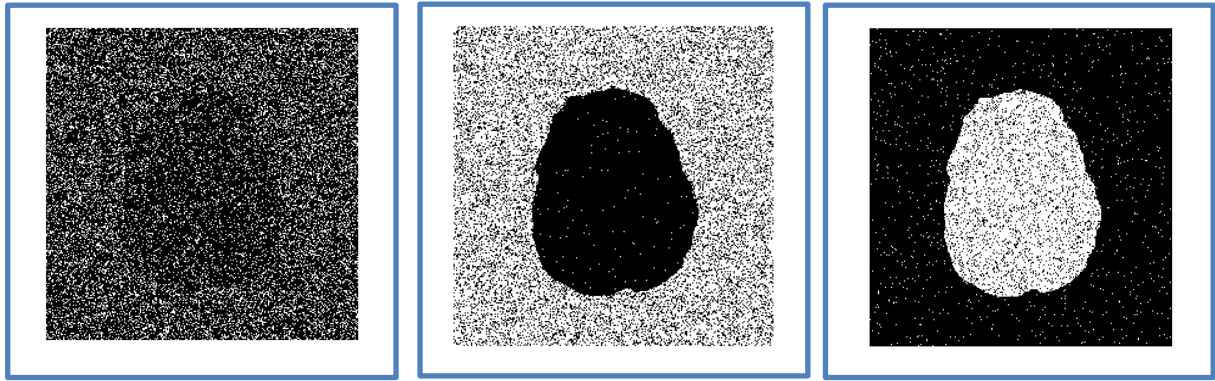


Figure 4.15: Classes de l'image IRM par l'approche FCM avec bruit gaussien 10% (images binaires)

Les résultats de la segmentation obtenus par l'approche FCM montrés dans la figure (4.11) visuellement, ainsi donne des meilleurs résultats en absence de bruit quand on les compare avec les images originales données dans la figure (4.4). Dans le cas où l'image est affectée, on remarque que cette approche ne peut pas classifier cette image, alors visuellement on dit que l'approche FCM donne mêmes résultats avec l'approche K-means. On constate que les deux approches ne sont pas robustes faces aux bruits.

#### IV.2.3. Approche neuronale RBF

L'approche neuronale c'est une approche intelligente basée sur les réseaux de neurones, dans nos travaux on s'est intéressé par le type RBF (Radial Basic Function) qui est présenté dans le chapitre 3.

La stratégie de cette approche s'articule autour de l'apprentissage à partir d'une base de données présélectionnée. La base de données est répartie en trois classes :

- Une classe de training : généralement c'est 4/9 de la base de données globale
- Une classe de validation : généralement c'est 2/9 de la base de données globale
- Une classe de test : le reste de la base qui est 1/3 du global (alors 2 trières pour le training et la validation, le trière pour le test).

Dans nos travaux, la base de données est présentée par l'image IRM cérébrale qui est de taille  $256*256=65536$  pixels :

- Training : on fait l'apprentissage par 1000 pixels aléatoires de l'image originale
- Validation : on valide par une autre mille pixels aléatoires de l'image originale
- Test : dans la phase de test, on considère l'image entière.

**Algorithme d'apprentissage :**

L'apprentissage est une phase du développement d'un réseau de neurone durant laquelle le comportement du réseau est modifié jusqu'à l'obtention du comportement désiré.

Souvent, l'algorithme d'apprentissage peut être décrit comme suit :

1. Déterminer les centres  $C_j$  des Gaussiens,
2. Calculer les largeurs  $\beta_j$  des Gaussiens de "kernels",
3. Calculer les poids  $\lambda_j$ .

Pendant les deux premières étapes seulement les entrées de l'ensemble de données de l'apprentissage sont employées, c'est pour cette raison qu'on l'appelle apprentissage non supervisé.

Les centres des Gaussiennes sont estimés selon un arrangement de quantification vectoriel, comme par exemple le compétitive learning. Une fois les centres et les largeurs sont calculés, le calcul des poids  $\lambda_j$  se fait par un apprentissage supervisé.

La fonction d'activation choisie est la fonction gaussienne qui est définie dans le chapitre 2. Pour la validation des résultats, un second ensemble appelé ensemble de test, est utilisé à la fin de chaque phase d'apprentissage. On mesure non seulement l'erreur d'apprentissage mais aussi l'erreur de test, c'est-à-dire l'erreur totale commise sur tous les exemples de l'ensemble de test. Cette erreur de test est calculée une fois que la phase d'optimisation des poids est terminée.

**Résultats d'extraction des classes :**

Pour valider l'approche proposée, on a introduire les fonctions Neural Network de Matlab. Pour créer un réseau RBF, on utilise la fonction : `newrb`.

Pour l'apprentissage du RBF, on utilise le code :

```
net = newrb(Xtrain',Ytrain',0,Beta,N_neur,N_neur)
```

Pour le test :

```
net = newrb(Xtrain',Ytrain',0,Beta_best,N_neur_best,N_neur_best);
```

Le nombre de neurone choisi varie entre 0 et 100, et la largeur de la gaussienne varie entre 0.1 et 10. Alors la stratégie est de chercher le nombre de neurone et la largeur optimaux qui donne une erreur minimale.

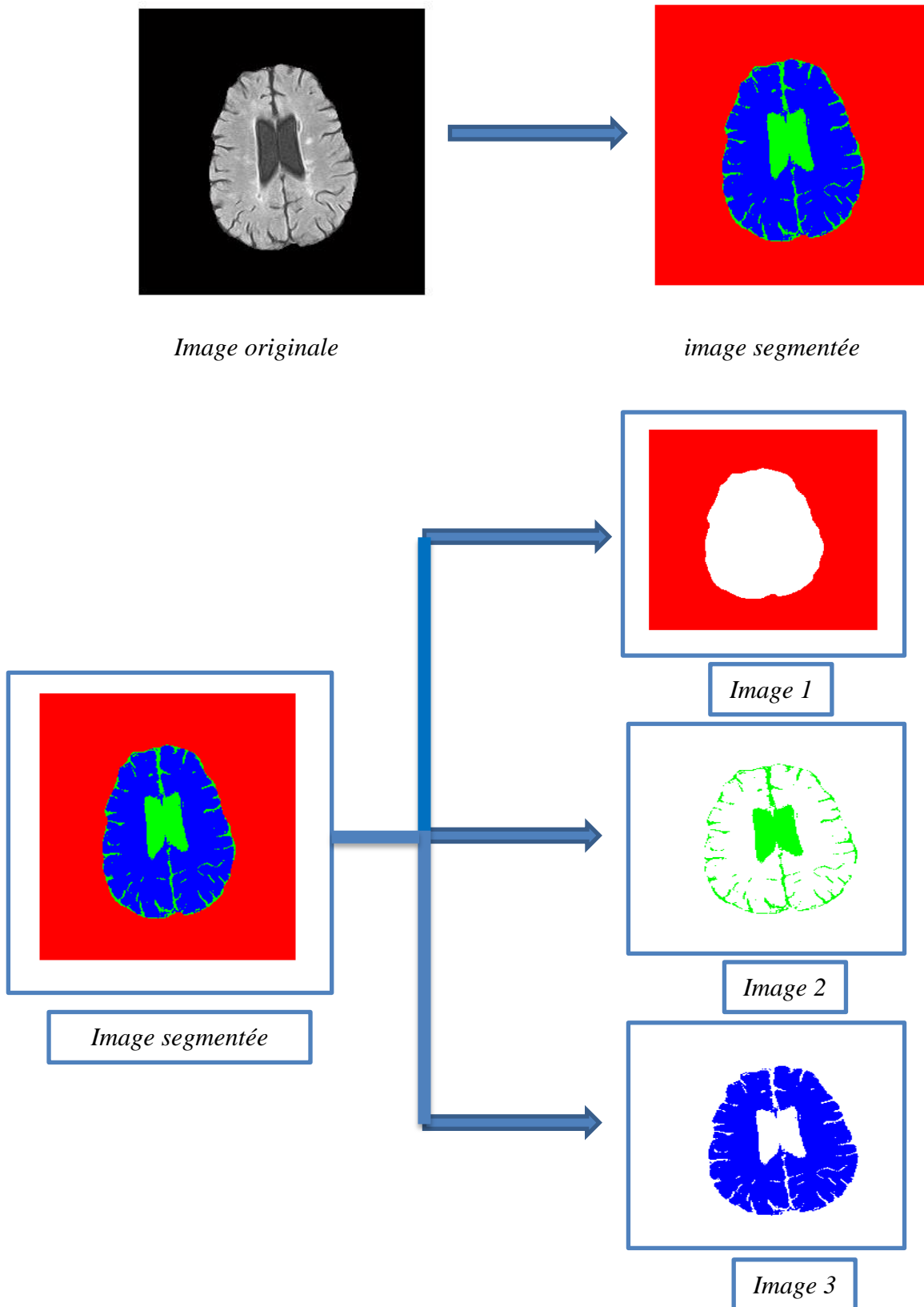


Figure 4.16 : Résultats de segmentation obtenus par l'approche neuronale RBF en absence du bruit



Figure 4.17: Classification par l'approche neuronale RBF sans bruit (images binaires)

A. Avec bruit gaussien (5%):

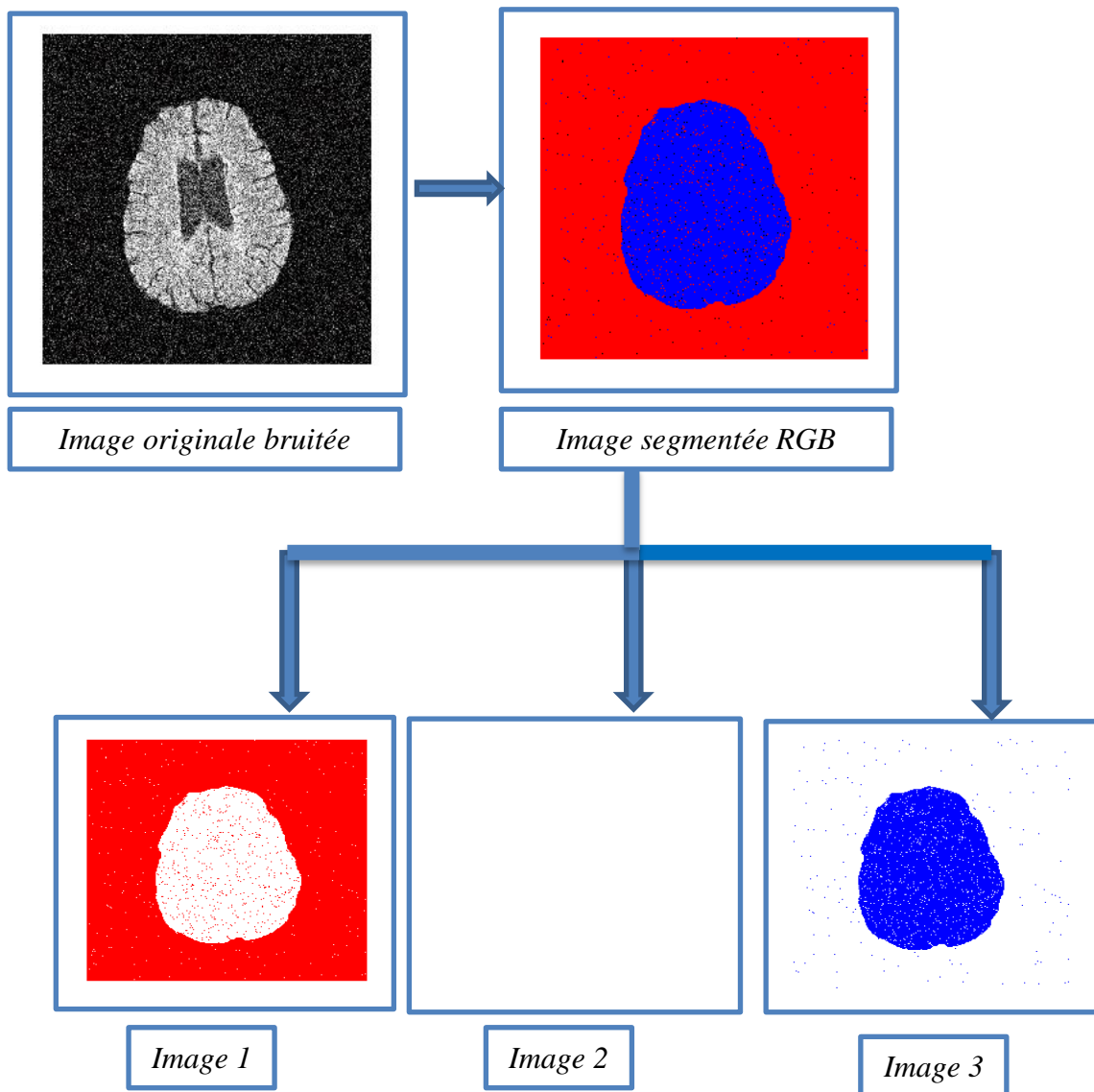


Figure 4.18: Résultats de segmentation par l'approche neuronale RBF en présence du bruit gaussien 5%



Figure 4.19: Classes de l'image IRM par l'approche neuronale RBF avec bruit gaussien 5% (images binaires)

**B. Avec bruit gaussien (10%):**

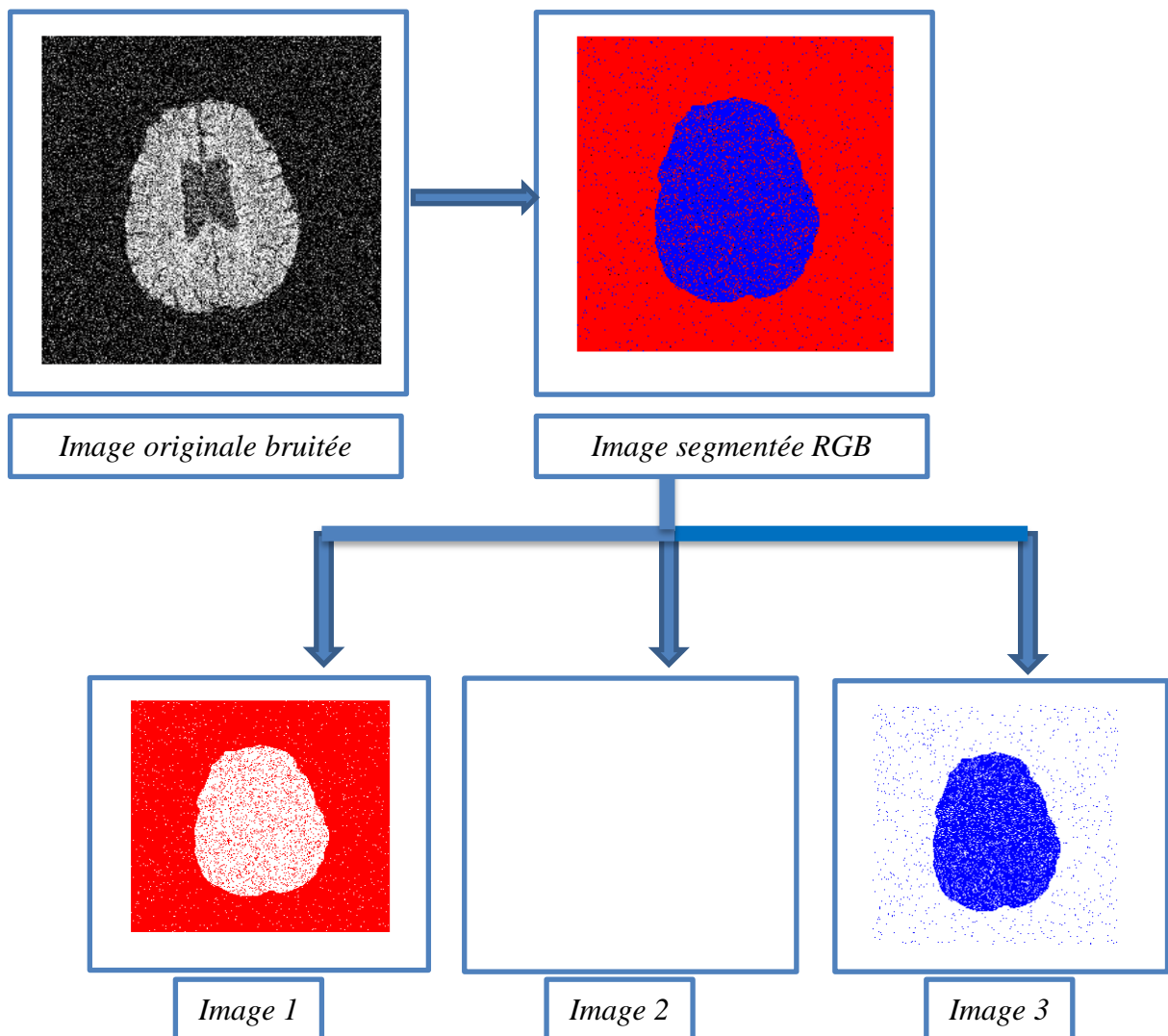


Figure 4.20: Résultats de segmentation par l'approche neuronale RBF en présence du bruit gaussien 10%

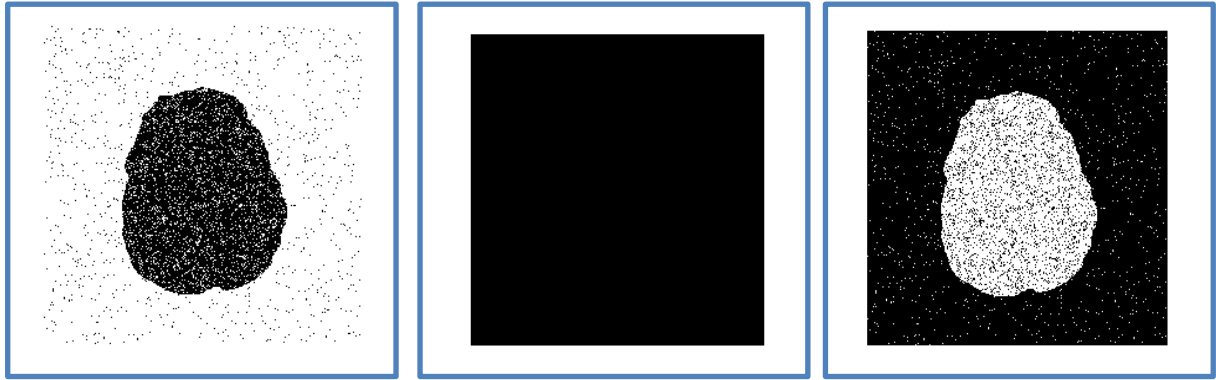


Figure 4.21: Classes de l'image IRM par l'approche neuronale RBF avec bruit gaussien 10% (images binaires)

Les résultats de la segmentation obtenus par l'approche neuronale RBF montrés dans la figure (4.17) visuellement, donne des meilleurs résultats en absence de bruit qui sont très proches à les deux approches précédentes quand on les compare avec les images originales données dans la figure (4.4). Les meilleurs nombre de neurones et la largeur  $\beta$  correspondent à une erreur minimale dans ce cas sont :

- $N=30$
- $B=4.4$

Dans le cas où l'image est affectée, on remarque que cette approche a des meilleures performances par rapport à les approches FCM et K-means. D'après la figure (4.19) où le cas de présence d'un bruit gaussien avec un taux de 5%, on remarque que les images obtenus sont proches aux images originales sauf que l'image 2 est une image blanche.

Pour le cas d'un bruit gaussien de 10%, les résultats obtenus sont très satisfaisants qui sont présentés par la figure (4.21). Les meilleurs nombre de neurones et la largeur  $\beta$  correspondent à une erreur minimale dans ce cas sont :

- $N=15$
- $B=0.3$

#### IV.2.4. Critère d'évaluation

Dans le but d'évaluer les résultats des approches de segmentation présentés dans ce travail, un critère d'évaluation est calculé pour chaque approche qui est le taux de reconnaissance :

$$Taux = \frac{Nb\text{re des pixels classés correctement}}{nb\text{re totale des pixels}} \times 100 \quad (4.1)$$

Ce taux correspond à la proportion de vrais positifs par rapport à l'ensemble des pixels qui devraient être segmentés. Cet indicateur permet d'évaluer dans quelle mesure l'intégralité d'une structure recherchée est segmentée.

Ces résultats sont présentés dans le tableau (4.1), nous avons calculé les taux de reconnaissance sur la totalité des images. Les résultats obtenus pour les trois approches obtenus sont comparés selon le taux de reconnaissance.

| Classe     | K-means |         |         | FCM     |         |         | Neuronale RBF |         |         |
|------------|---------|---------|---------|---------|---------|---------|---------------|---------|---------|
|            | 1       | 2       | 3       | 1       | 2       | 3       | 1             | 2       | 3       |
| Sans bruit | 99.8230 | 98.6404 | 98.8174 | 99.8230 | 98.6496 | 98.8266 | 99.8230       | 98.6403 | 98.8083 |
| Bruit 5%   | 42.1387 | 38.5315 | 93.5305 | 42.2913 | 38.5345 | 93.6401 | 98.6389       | 94.6701 | 93.6584 |
| Bruit 10%  | 5.5756  | 39.9035 | 63.8596 | 5.2414  | 39.7385 | 63.7909 | 94.8669       | 94.6701 | 90.9805 |

Tableau 4.1 : Taux de reconnaissance des approches

Nous avons mesuré et quantifié les performances de segmentation de l'ensemble des images en comparant les résultats avec les images vérité terrain. Pour cela nous avons utilisé un critère qui est le taux de reconnaissance. Des résultats visuels de la segmentation sont présentés.

Le tableau (4.1) montre les taux de reconnaissance obtenu pour les trois algorithmes K-means, FCM et neuronale sans bruit et avec bruit. Et d'après le tableau, on remarque que le taux de reconnaissance de l'algorithme FCM est plus élevé par rapport aux deux autres algorithmes pour les images sans bruits. FCM donne le meilleur taux de classification de 98.6496% (classe 2) pour les images sans bruit, et approche neuronale donne le meilleur de classification pour les images bruitées, avec bruit gaussien 5% et 10%, cet approche est plus robuste pour l'image affectée par un bruit de type gaussien.

A la lumière des résultats présentés et des résultats visuels, nous constatons une amélioration remarquable de la segmentation par l'approche neuronale RBF qui donne des meilleurs résultats qui sont montrés dans le tableau et visuellement.

### IV.3.Conclusion

Actuellement, la tendance en matière de segmentation d'image est de combiner deux ou plusieurs approches afin de tirer les avantages de chacune d'elles pour traiter au mieux la variabilité des images et la richesse d'information qu'elles contiennent.

Pour cela, nous avons proposé dans ce dernier chapitre de présenter trois approches de segmentation afin qu'on puisse extraire des meilleurs résultats qui reflètent les approches robustes. Finalement on a constaté que l'approche neuronale est très efficace même en présence de bruit.

---

---

## *Conclusion Générale*

---

---

## CONCLUSION GENERALE

Dans ce travail nous avons essayé de montrer l'importance des images médicales et leurs capacités à développer le monde la classification intéressante à la segmentation par approche classification, elle consiste à déterminer des procédures permettant de rassembler l'ensemble des pixels de l'image en un nombre limité de classes correspondant aux grands éléments structuraux de l'image. La classification peut se faire de deux manières :

La première suppose l'existence de certains pixels dont l'appartenance aux classes.

La seconde dit non supervisée, vise à regrouper automatiquement des pixels de l'image en classes.

Nous allons présenter un ensemble d'algorithmes qui répondent à cette dernière tel que les k-means et d'autres algorithmes qui s'appuient sur les concepts de la logique floue comme le c-moyenne floue ou Fuzzy C-Means (FCM) Un troisième algorithme connu sous le nom des FCM\_S, qui intègre la disposition spatiale des pixels. Ces techniques de classification seront appliquées sur des images médicales.

---

---

# ***Bibliographie***

---

---

# BIBLIOGRAPHIE

- 01- [ADA07] T. Adamek, N. O'Connorn, et A. Smeaton « Word matching using single closed contours for indexing handwritten historical documents » , IJDAR 2007
- 02- [ALF01] R. Alferez et Y.F. Wang. « Image indexing and retrieval using image-derived, Geometrically and Illumination Invariant Features. » IEEE Multimedia Conference, Japan, 2001.
- 03- [AME07] Z. Ameer, A. Adane, et S. Ameer « Etude comparative des méthodes de classification appliquée à la segmentation d'images texturées », SETIT 25/03/2007
- 04- [ARI03] N. ARICA « Shape : representation, description, similarity and recognition » thèse de docteur de l'université de the graduate school of natural and applied sciences , Septembre 2003
- 05- [BAR92] D. Barba et J. Benois « Image segmentation by region-contour cooperation for image coding » In ICPR92, La Haye, 1992.
- 06- [BEL02] S. Belongie, J. Malik, et J. Puzicha. « Shape matching and object recognition using shape contexts » IEEE Transactions on Pattern Analysis and Machine Intelligence, Avril 2002.
- 07- [BEL97] A. Belaïd, « Computer aided design of models of page for their use in recognition of documents » Workshop one Electronic Page Models LAMPE, 1997
- 08- [BER02] J. Bernd «Digital image processing», Springer 2002
- 09- [BIA 96] A BIANCARDI et A. MÉRIGOT «Connected Component Support for Image Analysis Programs », International Conference on Pattern Recognition ICPR, 1996.