



MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE
SCIENTIFIQUE

UNIVERSITÉ ABDELHAMID IBN BADIS - MOSTAGANEM



Faculté des Sciences Exactes et d'Informatique
Département de Mathématiques et informatique
Filière : Informatique

MÉMOIRE DE FIN D'ÉTUDES

Pour l'Obtention du Diplôme de Master en Informatique
Option : **Ingénierie des Systèmes d'Information**

THÈME :

Détection automatique de la maladie de parkinson Via
des algorithmes bio-inspirés

Étudiant(e) : « **SAID Fatma** »

« **OUAHRANI Ghania Lamia** »

Encadrant(e) : « **Monsieur BESNASSI Miloud** »

Année Universitaire 2022-2023

Dédicaces

Tout d'abord je tiens à remercier **ALLAH** de m'avoir aidé à arriver jusqu' à ce niveau d'études, et de réaliser ce modeste travail, que je dédie :

À ma chère mère KERABCHI Aicha

À mon père Hadj Mansour

À ma chère sœur Noria

Qui n'ont jamais cessé, de formuler des prières à mon égard, de me soutenir et de m'épauler pour que je puisse atteindre mes objectifs.

À mon frère Allah yarhamh KERABCHI Mehdi

À mes frères KERABCHI Hamza, KERABCHI Mohamed, KERABCHI Kacem

Pour ses soutiens moraux et leurs conseils précieux tout au long de mes études.

Sans oublier ma chère binôme OUAHRANI Gania Lamia

Pour son entente et sa sympathie.

À toute la promotion 2023.

Enfin à toutes personnes qui m'ont soutenu et aidé à réaliser ce travail

Mme : SAID Fatma

À mon cher père HAKIM,

Pour son soutien, son affection et la confiance qu'il m'a accordé et de m'encourager durant ces années d'études.

À ma chère maman,

Pour son amour, son encouragement, et de me m'épauler pour que je puisse atteindre mes objectifs.

À mon frère YASSINE, et à mes chères sœurs LINA et SARA, et ma chère tante NORIA,

Pour ses soutiens moraux et de me supporter dans des moments difficiles.

Que dieu les protège et leurs offre la chance et le bonheur.

Sans oublier ma chère binôme SAID FATMA,

Pour son entente, sa patience et sa compréhension tout au long de ce projet.

OUAHRANI Ghania Lamia

Remerciements

Tout d'abord je tiens remercier **ALLAH** le tout puissant de m'avoir donné la santé, la volonté, le courage et la patience pour mener à terme ma formation et pourvoir réaliser ce travail de recherche.

Nous tenons à adresser nos sincères remerciements à notre encadreur Monsieur **BESNASSI Miloud** pour son encadrement, sa disponibilité, son aide qui nous avons amenées à réaliser ce modeste travail. Nous avons beaucoup appris.
Merci encore.

Nous remercions également Monsieur **LAREDJ Mohamed Adnane** et Madame **HAMAMI Dalila** d'avoir accepté d'examiner notre travail.

Résumé

La maladie de Parkinson est une maladie chronique qui touche généralement les personnes âgées de plus de 60 ans, mais peut également toucher les personnes plus jeunes. Le tremblement des mains est le symptôme le plus fréquemment utilisé pour diagnostiquer cette maladie. Les algorithmes bio-inspirés sont un nouveau domaine de recherche utilisé, la plupart des études précédentes se focalisent sur un algorithme CNN appliqué sur une base des données (HandPD).

Durant ce travail nous avons pris base des données (HandPD) pour la tester avec d'autres algorithmes, afin d'améliorer le temps d'exécution et le taux de reconnaissance.

Nous avons suivi trois volets principaux, le premier s'intéresse au module de détection de la maladie et l'extraction des zones pertinentes des images, en utilisant quatre descripteurs : les motifs binaires locaux (LBP), LBP-52, l'histogramme de gradient orienté (HOG) et la matrice de co-occurrence en niveau de gris (GLCM). Le deuxième axe porte des méthodes de classification (KNN-MLP-SVM). Le troisième, les techniques d'optimisation avancées dont le but est de réduire la dimension des vecteurs des caractéristiques. La sélection des zones a été réalisée à l'aide de plusieurs inspirations comme la théorie de l'intelligence distribuée, les fonctions mathématiques et les lois de la physique. Ensuite une comparaison finale entre les résultats de la détection obtenus et celles des études précédentes, afin de confirmer notre hypothèse.

Mots-clés :

Maladie Parkinson, Méthodes extraction, Symptôme, Tremblement, rigidité, Histogramme de gradient orienté, Les descripteurs locaux LBP, La matrice de cooccurrence en niveau de gris (GLCM), méta-heuristique, bio-inspirée, KNN (K-Nearest Neighbors) et MLP (Multi-Layers Perceptron).

Abstract

Parkinson's disease is a chronic disease that usually affects people over the age of 60, but can also affect younger people. Hand tremor is the most frequently used symptom to diagnose this disease. Bio-inspired algorithms are a new area of research used, most previous studies focus on a CNN algorithm applied to a database (HandPD).

During this work we took the database (HandPD) to test it with other algorithms, in order to improve the execution time and the recognition rate.

We followed three main parts, the first focuses on the disease detection module and the extraction of the relevant areas of the images, using four descriptors: the local binary patterns (LBP), LBP-52, the histogram of oriented gradient (HOG) and the grayscale co-occurrence matrix (GLCM). The second axis concerns classification methods (KNN-MLP-SVM). The third, advanced optimization techniques whose goal is to reduce the dimension of feature vectors. The selection of the zones was carried out using several inspirations such as the theory of distributed intelligence, mathematical functions and the laws of physics. Then a final comparison between the detection results obtained and those of previous studies, in order to confirm our hypothesis.

Keywords:

Parkinson's disease, Methods extraction, Symptom, Tremor, rigidity, Oriented gradient histogram, LBP (local descriptors), Gray scale co-occurrence matrix (GLCM), metaheuristic, bio-inspired, KNN (K-Nearest Neighbors) and MLP (Multilayer Perceptron).

Liste des figures

Figure N°	Titre de la figure	Page
Figure 1	circuits moteurs primaires dans les ganglions	19
Figure 2	Exemple de matrice de co-occurrence avec $d = 1$ et $\theta = 0^\circ$	31
Figure 3	Le code LBP original	35
Figure 4	LBP multi-échelle	37
Figure 5	Construction et uniformité d'un motif LBP	41
Figure 6	Représentation d'une image spirale par les histogrammes du code LBP	41
Figure 7	Principe de la décomposition pour le calcul des HOG	45
Figure 8	Exemple d'un perceptron multicouche	48
Figure 9	Comparaison d'hyperplans avec différents noyaux	55
Figure 10	Effet du coefficient de régularisation	57
Figure 11	La hiérarchie sociale des loups gris	64
Figure 12	Encerclement de la proie	67

Figure 13	Les différentes phases de HHO	68
Figure 14	Le processus d'encerclement fort avec une descente progressive rapide dans un espace à 2D	70
Figure 15	La stratégie de Lévy (MPA)	77
Figure 16	Prétraitement des images	78
Figure 17	les résultats des mesures de performances de chaque descripteurs	85
Figure 18	Résultats de LBP-52 en terme de fitness	87
Figure 19	Résultats de LBP-52 en terme de fitness	88
Figure 20	Résultats de HOG en terme de fitness	88
Figure 21	Résultats de GLCM en terme de fitness	89
Figure 22	Résultats de LBP en terme d'accuracy	91
Figure 23	Résultats de LBP-52 en terme d'accuracy	91
Figure 24	Résultats de HOG en terme d'accuracy	92
Figure 25	les résultats des mesures de performances de chaque descripteurs	92

Figure 26	la comparaison des algorithmes en terme de fitness	93
Figure 27	Résultats de GWO en terme de fitness	95
Figure 28	Résultats de HHO en terme de fitness	95
Figure 29	Résultats de AOA en terme de fitness	96
Figure 30	Résultats de MPA en terme de fitness	96
Figure 31	Résultats de GWO en terme d'accuracy	98
Figure 32	Résultats de HHO en terme d'accuracy	98
Figure 33	Résultats de AOA en terme d'accuracy	99
Figure 34	Résultats de MPA en terme d'accuracy	99
Figure 35	les résultats des mesures de performances de chaque classifieurs	101

Liste des abréviations

Abréviation	Expression Complète
MP	Maladie parkinson
IRM	Imagerie par résonance magnétique
SPECT	Tomographie d'émission mono photonique
PET	Tomographie par émission de positons
EEG	Électro-encéphalographie
PD	Parkinson's disease
AOA	Archimedes Optimization Algorithm
GWO	Grey Wolf Optimization
GLCM	Gray Level Co-occurrence Matrix
MPA	Algorithme des prédateurs marins

Liste des tableaux

Tableau N°	Titre du tableau	Page
Tableau 1	Matrice de confusion	82
Tableau 2	Résultats des descripteurs pour chaque algorithme en terme de fitness	86
Tableau 3	Résultats des descripteurs pour chaque algorithme en termes d'accuracy	89

Table des matières

Introduction Générale.....	15
Chapitre 1 Généralités	17
1.1 Introduction	17
1.2 Mise en contexte.....	17
1.3 La maladie de parkinson	17
1.3.1 Physiopathologie	18
1.3.2 Symptomatologie de la maladie de Parkinson	19
1.4 Méthodes diagnostiques	21
1.5 État de l’art	22
1.6 Conclusion.....	26
Chapitre 2 Les méthodes d’extraction et les approches de classification	27
2.1 Introduction	27
2.2 Extraction des caractéristiques	27
2.3 Les méthodes d’extraction	28
2.3.1 La matrice de Co-occurrence à niveau de gris (GLCM).....	28
2.3.2 Motifs binaires locaux (LBP).....	34
2.3.3 Histogramme à base du gradient orienté (HOG).....	42
2.4 Méthodes de classification	45
2.4.1 Le modèle de Perceptron Multi-Couches (MLP).....	45
2.4.1.1 Architecture.....	46
2.4.1.2 Rétro-propagation du gradient.....	48

2.4.1.3	Avantages du perceptron à une couche cachée	51
2.4.2	Méthode des k plus proches voisins (KNN).....	52
2.4.3	Méthodes d'apprentissage machine classique	55
2.4.3.1	Machine à vecteurs de support (SVM)	55
2.5	Conclusion.....	57
Chapitre 3 Les techniques d'optimisation avancées		58
3.1	Introduction	58
3.2	Algorithmes évolutionnaires	58
3.2.1	Principe des algorithmes évolutionnaires	59
3.2.2	Mise en œuvre des algorithmes évolutionnaires	60
3.2.2.1	Le codage ou représentation.....	61
3.3	Les différents types d'algorithmes évolutionnaires	62
3.3.1	Les algorithmes inspirés de l'intelligence distribuée.....	62
3.3.1.1	L'algorithme des loups gris (GWO).....	62
3.3.1.2	L'algorithme des faucons (HHO).....	67
3.3.2	Les algorithmes inspirés de la physique	71
3.3.2.1	Algorithme d'optimisation d'Archimède (AOA).....	71
3.3.2.2	Algorithme des prédateurs marins (MPA)	75
3.3.3	Structure de l'objet immergé	77
3.4	Conclusion.....	79
Chapitre 4 Résultats Expérimentaux & Discussion.....		80
4.1	Introduction	80

4.2	Les mesures de performance	81
4.3	Analyse graphique	85
4.3.1	Résultats des descripteurs pour chaque algorithme	86
4.3.1.1	En terme de fitness et taux de reconnaissance	86
4.3.2	Résultats des algorithmes pour chaque descripteur	94
4.3.2.1	En terme de fitness et taux de reconnaissance	94
4.3.3	Résultats pour les deux classifieurs (MLP, SVM)	100
4.4	Conclusion.....	102
	Conclusion Générale	103
	Bibliographie.....	105

Introduction Générale

La maladie de Parkinson est une affection chronique neurodégénérative, elle se caractérise par la disparition progressive de certains neurones dans le cerveau.

Au fil des années, des progrès ont été réalisés dans la compréhension de la maladie et des nouvelles approches de diagnostic et de traitement ont été développées. Parmi ces approches, l'utilisation d'algorithmes bio-inspirés pour la détection automatique qui peut offrir de nouvelles perspectives pour la recherche et la pratique clinique.

Ce travail cherche à introduire un nouveau concept de la détection automatique avec des techniques d'optimisation avancées.

Problématique :

La détection automatique de la maladie de Parkinson via des algorithmes bio-inspirés nécessite la phase d'extraction des zones à partir des images de tremblement des mains contenant des dessins (spirale) de plusieurs individus (HandPD), d'après les études précédentes, le CNN est l'algorithme le plus utilisé malgré ses inconvénients, tandis qu'il y a d'autres algorithmes plus performants. Alors quels sont ces algorithmes ?

Structure de mémoire :

Ce mémoire est structurée de la manière suivante :

L'introduction générale, définit le contexte général de ce travail et ces objectifs.

Le premier chapitre présente la généralité sur la maladie de parkinson et ces symptômes.

Le deuxième chapitre introduire les méthodes d'extractions des caractéristiques à base de l'histogramme gradient orienté, (HOG), les motifs binaires locaux (LBP), (LBP-52) et la matrice des cooccurrences en niveau de gris (GLCM). Les modèles de classification comme les réseaux de neurones et les méthodes de classification à base de distance.

Le troisième chapitre résume les techniques d'optimisation incluant les différentes inspirations, issues de la théorie d'intelligence distribuée dans le cadre des algorithmes : GWO (Grey Wolf Optimizer), HHO (Harris Hawks Optimizer), MPA (Marine Predators Algorithm) les lois de la physique dans le cadre des algorithmes AOA (Archimedes Optimization).

Le quatrième chapitre traite les métriques utilisées, l'étude statistique plus les résultats obtenus de la détection de la maladie en utilisant les algorithmes avancées AOA, MPA, HHO, GWO avec le classifieur KNN, avec les quatre méthodes d'extraction GLCM, LBP, LBP-52, HOG, après l'optimisation on obtient le meilleur résultat pour les différents algorithmes, ensuite on va appliquer les classificateurs MLP, SVM avec les descripteurs et les algorithmes plus performants. Enfin une comparaison entre les résultats des algorithmes obtenus et celles des études précédentes.

Chapitre 1

Généralités

1.1 Introduction

Ce chapitre présente essentiellement les informations générales sur la maladie de parkinson. Dans un premier temps, on va décrire c'est quoi la maladie de parkinson, puis les Symptomatologie de la maladie, ensuite les méthodes pour diagnostiquer cette maladie, et on va donner un aperçu sur les travaux précédents.

1.2 Mise en contexte

Le développement d'outils de diagnostic requièrent d'abord une compréhension des maladies traitées. La présente section est consacrée à une présentation de la maladie qui sera traitée dans ce projet.

1.3 La maladie de parkinson

La maladie de Parkinson est la deuxième maladie neurodégénératives la plus fréquente après l'Alzheimer [1]. Les origines de la maladie ne sont pas encore bien précises. Des études suggèrent des causes génétiques [2], alors que d'autres des causes environnementales [3]. la maladie de Parkinson résulte d'une dégénération des neurones dopaminergiques du cerveau qui entraîne un dérèglement du système dopaminergique [4].

1.3.1 Physiopathologie

La physiopathologie de la maladie de Parkinson implique la dégénérescence progressive des neurones dopaminergiques dans la substantia nigra, une région du cerveau impliquée dans la régulation des mouvements et des émotions. Les neurones dopaminergiques produisent de la dopamine, un neurotransmetteur qui aide à transmettre les signaux entre les cellules nerveuses. Au fur et à mesure que ces neurones meurent, il y a une diminution progressive de la production de dopamine dans le cerveau, ce qui entraîne une perturbation de la communication entre les cellules nerveuses et la perte de contrôle des mouvements. En plus de la dégénérescence des neurones dopaminergiques, il y a également une accumulation de protéines anormales, appelées corps de Lewy, dans le cerveau. Ces corps de Lewy se trouvent principalement dans les neurones dopaminergiques, mais ils peuvent également être présents dans d'autres régions du cerveau. L'accumulation de corps de Lewy dans le cerveau est associée à des symptômes non moteurs de la maladie de Parkinson, tels que la dépression, l'anxiété et les troubles du sommeil. Les mécanismes exacts de la formation de ces corps de Lewy et de leur rôle dans la maladie de Parkinson ne sont pas encore entièrement compris. Enfin, des facteurs génétiques et environnementaux peuvent également contribuer au développement de la maladie de Parkinson. Les mutations génétiques dans certains gènes, tels que le gène LRRK2, ont été associées à une augmentation du risque de développer la maladie de Parkinson. De même, l'exposition à des toxines environnementales, telles que le pesticide roténone et le solvant trichloréthylène, a été associée à un risque accru de développer la maladie de Parkinson.

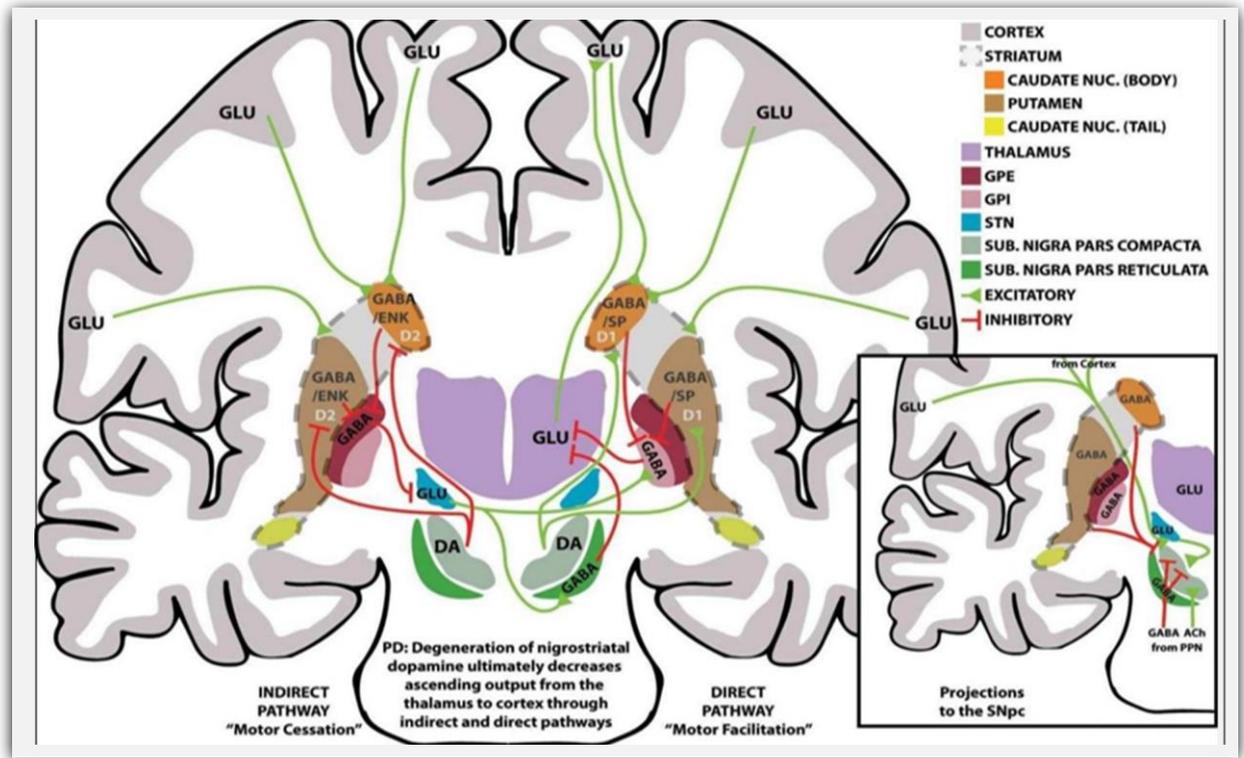


Figure 1 - circuits moteurs primaires dans les ganglions [5]

1.3.2 Symptomatologie de la maladie de Parkinson

La maladie de Parkinson se caractérise par des symptômes moteurs et non moteurs, qui peuvent varier d'une personne à l'autre en termes de sévérité et de progression. Des symptômes les plus courants de la maladie de parkinson :

1. Les symptômes moteurs de la maladie de Parkinson sont causés par la dégénérescence progressive des neurones producteurs de dopamine dans le cerveau, ce qui affecte la capacité

de l'organisme à contrôler les mouvements. Voici quelques symptômes moteurs les plus courants de la maladie de Parkinson :

- Tremblements : mouvements involontaires, généralement au repos, des mains, des bras, des jambes, de la tête ou du menton.
- Rigidité musculaire : raideur musculaire qui rend les mouvements difficiles et peut causer de la douleur.
- Bradykinésie : ralentissement des mouvements, qui peut rendre les activités quotidiennes plus difficiles.
- Instabilité posturale : difficulté à maintenir l'équilibre et à se tenir debout, ce qui peut augmenter le risque de chutes.
- Hypokinésie faciale : diminution de l'expression du visage, qui peut donner l'impression d'un visage figé.

2. La maladie de Parkinson peut également causer une variété de symptômes non moteurs qui peuvent avoir un impact important sur la qualité de vie des patients. Voici quelques symptômes non moteurs les plus courants de la maladie de Parkinson :

- Troubles du sommeil : difficultés à s'endormir, somnolence diurne excessive, insomnie.
- Constipation : diminution de la fréquence des selles.
- Perte de l'odorat : diminution de la capacité à sentir les odeurs.

- Troubles de la mémoire et de la cognition : difficultés à se concentrer, à mémoriser et à raisonner.

1.4 Méthodes diagnostiques

La maladie de parkinson est souvent basée sur l'observation des symptômes moteurs et non moteurs du patient. Cependant, les médecins peuvent utiliser plusieurs méthodes diagnostiques pour confirmer et éliminer d'autres conditions qui pourraient causer des symptômes similaires qui sont :

- Examen neurologique : un examen approfondi des mouvements, des réflexes, de la coordination et de la force musculaire du patient peut aider à identifier les symptômes moteurs caractéristiques de la maladie de Parkinson.
- Tomographie par émission de positons (TEP) : cette technique d'imagerie peut aider à identifier les zones du cerveau qui sont affectées par la maladie de Parkinson en mesurant l'activité métabolique des cellules cérébrales.
- Imagerie par résonance magnétique (IRM) : cette technique d'imagerie peut être utilisée pour exclure d'autres affections cérébrales qui pourraient causer des symptômes similaires à ceux de la maladie de Parkinson.
- Test de la levodopa : ce test consiste à administrer une dose de levodopa, un médicament qui augmente les niveaux de dopamine dans le cerveau, pour voir si les symptômes moteurs du patient s'améliorent. Si les symptômes s'améliorent, cela peut indiquer que la personne est atteinte de la maladie de Parkinson.

- Tests génétiques : bien que la maladie de Parkinson ne soit pas strictement héréditaire, certains gènes ont été associés à un risque accru de développer la maladie de Parkinson, et les tests génétiques peuvent être utilisés pour identifier ces gènes.

1.5 État de l'art

Le tremblement des mains est le symptôme le plus fréquemment utilisé par les patients parkinsoniens pour détecter le trouble à partir d'images / croquis / écritures dessinés à la main, ainsi que de nombreux autres symptômes de la maladie de Parkinson. On va citer certains travaux :

(Sakar et al,2013) Dans leur recherche, les données ont été obtenues à partir de 40 personnes, 20 étaient en bonne santé et 20 étaient des patients malades subjectifs. Un total de 26 discours des échantillons a été prélevés sur chaque individu, y compris des phrases, des mots et des chiffres. Pour la classification et la validation croisée, ils ont utilisé SVM et KNN. Le classificateur KNN a produit une précision de 82,50 %, alors que le classificateur SVM a rapporté une précision de 85 % [6].

(Pereira et al,2015), ils ont utilisé des algorithmes d'apprentissage non profonds pour diagnostiquer la MP. Leurs recherches visant à : construire un jeu de données public appelé "HandPD", ensuite créer un pipeline pour le traitement d'images pour extraire certaines caractéristiques des images, puis présenter une nouvelle métrique quantitative appelée « Mean Tremblement relatif » pour mesurer le tremblement d'un individu. Ils ont mis l'accent sur deux tâches principales : traitement d'images et extraction de caractéristiques. La tâche de traitement

d'image contient un pipeline de plusieurs étapes pour extraire le gabarit en spirale ST et la trace manuscrite HT, car l'image comprenait le modèle que le sujet a été invité à suivre. Extraction de caractéristiques vise à évaluer les différences entre le ST et le HT en calculant neuf valeurs statistiques entre eux. Ils ont utilisé trois modèles de classificateurs dans leurs expériences : naïve Bayes (NB), Optimum-Path Forest (OPF) et SVM. Ils ont construit un ensemble de données appelé « HandPD » à partir de 55 individus : 37 (25 hommes et 12 femmes) patients et 18 (6 hommes et 12 femmes) témoins sains. Cet ensemble de données contient un test manuscrit images pour spirale et méandre, mais dans leurs recherches, ils se sont concentrés uniquement sur la spirale. Dans tout, ils ont utilisé 373 échantillons, car ils ont environ quatre spirales par individu et par examen. Ils se sont divisés ces échantillons à 90 % de formation et 10 % en tant qu'ensemble de test.

(Drotar et al,2016) a utilisé la Base de données (PaHaW), qui comprenait 37 patients parkinsoniens et 38 en bonne santé les personnes qui ont effectué huit tâches d'écriture différentes, comme dessiner un Archimède écrire en spirale et orthographiquement des syllabes, et des phrases. Cette étude utilisé trois classificateurs le plus proche voisin (KNN), la machine à vecteurs de support (SVM) et AdaBoost pour prédire la MP à l'aide de données conventionnelles de pression cinématique et d'écriture manuscrite, avec une précision de 81,3 % [7].

(Benba et al, 2016), ils ont utilisé un ensemble de données contenant 34 voyelles persistantes, provenant de 34 individus, dont 17 étaient subjectives patients et 17 étaient en bonne santé. Pour la classification, la technique SVM a été utilisée, quel a obtenu un taux de précision de 91,17% en utilisant les douze premiers coefficients de la Fréquence Mel Coefficients cestreaux (MFCC) par noyau SVM [8].

(Choi et al,2017), ils ont utilisé leur modèle proposé (MP net) avec deux ensembles de données : l'ensemble de données PPMI, qui a obtenu une précision de 96 %, et un ensemble de données privé (cohorte SNUH) avec une précision de 98,8 % [9].

(Haq et al, 2019), ils ont utilisé un ensemble de données de la voix des patients, la plus grande précision pour diagnostiquer la MP a été obtenu en utilisant SVM L1-Norm avec validation croisée K-fold ; K = 10 ayant un taux de précision de 99 % [10]. La précision minimale a été obtenue à partir de naïve Bayes, avec un taux de précision de 70,26 %.

(Vasquez et al,2019), ils ont proposé un modèle CNN avec une base des données (HandPD) avec une précision de 99,5 % [11] .

(Avuçlu et al,2020), ils ont examiné la détection des sujets MP à l'aide de diverses techniques de ML. Ils ont mené leur expérience à la fois sur des données de formation et de test, où ils ont utilisé 22 caractéristiques acoustiques de 195 enregistrements sonores. Pour diagnostiquer la MP, quatre classificateurs d'apprentissage automatique ont été utilisé : KNN, SVM, Naïve Bayes et forêt aléatoire. L'algorithme Naïve Bayes a diagnostiqué des patients atteints de MP avec une précision de 70,26 % et une précision de 64 % pour les données de test [12].

(Ozsahin et al, 2020), ils ont proposé un réseau de neurones à rétro propagation (BPNN), qui atteint une précision de 99,6 % [13].

(Ali et al, 2020) qui ont proposé un réseau de neurones génétiquement optimisée (GONN) avec une précision du modèle de 100 % [14]. À l'heure actuelle, davantage d'études ont soutenu le modèle CNN pour l'analyse de la parole.

(Diaz et al, 2021), Ils ont proposé un réseau de neurones à rétro propagation CNN-RNN avec des bases des données (PaHaW) [15], (NewHandPD) [16] qui atteint une précision de 90.0 %.

1.6 Conclusion

Dans ce chapitre, on a introduit un survol rapide de quelques travaux portant sur la détection automatique de la maladie de parkinson.

Il est important de noter qu'il y'a un nombre limité de travaux de recherche qui traitent la détection de la maladie de parkinson à base des algorithmes bio-inspirée.

Dans le chapitre suivant, on va exposer les procédures liées au prétraitement. En premier lieu, la description des méthodes d'extraction des caractéristiques comme la méthode des motifs binaires locaux (LBP), LBP-52, l'histogramme du gradient orienté (HOG) et la matrice de Co-occurrence en niveau de gris. En second lieu, la description des types de classifieurs comme le perceptron multicouches (MLP) et les classifieurs à base de distance k-NN ,SVM.

Chapitre 2

Les méthodes d'extraction et les approches de classification

2.1 Introduction

De nombreuses méthodes ont été proposées pour décrire la caractéristique de texture. On va présenter ci-dessous une brève étude de quelques techniques existantes sur l'extraction des caractéristiques de texture appliquées à l'analyse des images. L'objectif de l'extraction des caractéristiques dans le domaine de la reconnaissance des formes consiste à exprimer les primitives sous une forme numérique ou symbolique appelée codage. Dans cette partie, on va introduire les descripteurs qui seront utilisés dans la partie expérimentations et résultats. Il s'agit tout d'abord des motifs binaires locaux (LBP),(LBP-52), puis les descripteurs d'histogramme à base de gradient orienté (HOG), et enfin la matrice de cooccurrence en niveaux de gris (GLCM).

2.2 Extraction des caractéristiques

L'une des étapes les plus importantes de l'approche traditionnelle de l'apprentissage est l'extraction des caractéristiques. La précision de la classification dépend principalement des caractéristiques extraites. L'extraction des caractéristiques est un processus au cours duquel les données riches redondantes sont extraites en se basant sur l'information de bord, la texture, la

couleur et l'information de frontière de l'image donnée. Les caractéristiques de la matrice de cooccurrence (GLCM) forment les caractéristiques de texture. Les caractéristiques de haut niveau extraites sont l'histogramme orienté gradient (HOG) et le motif binaire local (LBP). L'extraction des caractéristiques est une composante du traitement de l'image. En effet, pour établir une règle de classification (supervisée ou non), on se base généralement sur un ensemble de critères numériques décrivant l'objet ou le phénomène observé $x \in X$.

2.3 Les méthodes d'extraction

2.3.1 La matrice de Co-occurrence à niveau de gris (GLCM)

L'une des méthodes les plus utilisées pour décrire la caractéristique de texture est la matrice de cooccurrence à niveau de gris (Gray Level Cooccurrence Matrix (GLCM)). Introduite par (Haralick) en 1973 [17], la matrice de cooccurrence permet de déterminer la fréquence d'apparition des paires de valeurs de pixels situés à une certaine distance dans l'image. Cette méthode considère la relation entre deux pixels voisins, le premier pixel est connu comme une référence et le second est connu comme un pixel voisin.

- **Principe :** Le calcul d'une matrice de cooccurrence de niveaux de gris ou GLCM (Grey Level Cooccurrence Matrix) consiste à repérer dans une image le nombre d'occurrences de paires de niveaux de gris séparés par une distance d dans une direction définie par un vecteur de déplacement (dx, dy) . Le calcul de la GLCM pour une image I de dimension $N * M$ se formalise comme suit :

$$GLCM_{dx,dy}(i,j) = \sum_{n=1}^N \sum_{m=1}^M \begin{cases} 1 & \text{Si } I(n, m)=i \text{ et } I(n + dx, m + dy)=j \\ 0 & \text{Sinon} \end{cases}$$

2.1

Où i et j sont les niveaux de gris du pixel de référence et du pixel voisin respectivement, n et m correspondent aux coordonnées des pixels dans l'image I . On peut déduire de **l'équation 2.1** que les entrées situées à proximité de la diagonale de la matrice de cooccurrence seront d'autant plus élevées que les valeurs de niveaux de gris de l'image sont proches ou similaires pour le déplacement considéré. A l'inverse, les entrées hors diagonale seront plus élevées pour des images dans lesquelles les niveaux de gris varient localement pour le déplacement considéré.

De même, le contenu textural s'exprimera différemment en fonction de la distance et de l'orientation du déplacement considéré entre les paires de sites. En pratique, les entrées de la GLCM sont souvent normalisées en les divisant par le nombre de pixels (somme des entrées égale à 1) afin d'éliminer l'influence de la taille de l'image. En outre, la matrice de cooccurrence est généralement construite de manière symétrique en considérant la somme des entrées calculées pour deux déplacements de directions opposées. Ainsi, pour un déplacement horizontal sur une distance de 1, on intégrera simultanément les occurrences obtenues pour deux vecteurs de déplacement (dx, dy) égal à $(1,0)$ et à $(0,1)$. Notons également l'existence d'une forme généralisée des matrices de cooccurrence dans laquelle les paires d'occurrences sont déterminées à partir des maxima locaux et de l'orientation des gradients autour de ces maxima. Cette version généralisée est particulièrement adaptée à la caractérisation de macro textures [18].

La formation des matrices de cooccurrence est une étape initiale qui compile des informations spatiales et statistiques pour calculer les paramètres de texture. L'information spatiale considérée est la position relative des paires de pixels, définies avec la distance d et les orientations Θ qui décrivent l'emplacement du second pixel par rapport à la première. Une

matrice de cooccurrence est formée pour chaque position. Typiquement, quatre valeurs de Θ , à savoir 0, 45, 90 et 135, et le choix de distance le plus courant est $d = 1$ est lorsque Θ est 0 ou 90, $d = 2$ quand Θ est 45 ou 135.

La figure 2 illustre la configuration de 4 GLCM qui correspondent aux 4 directions avec la distance fixée à $d = 1$. Les caractéristiques extraites des GLCM incluent principalement l'énergie, le contraste, l'entropie, la corrélation, l'homogénéité, et la dissemblance.

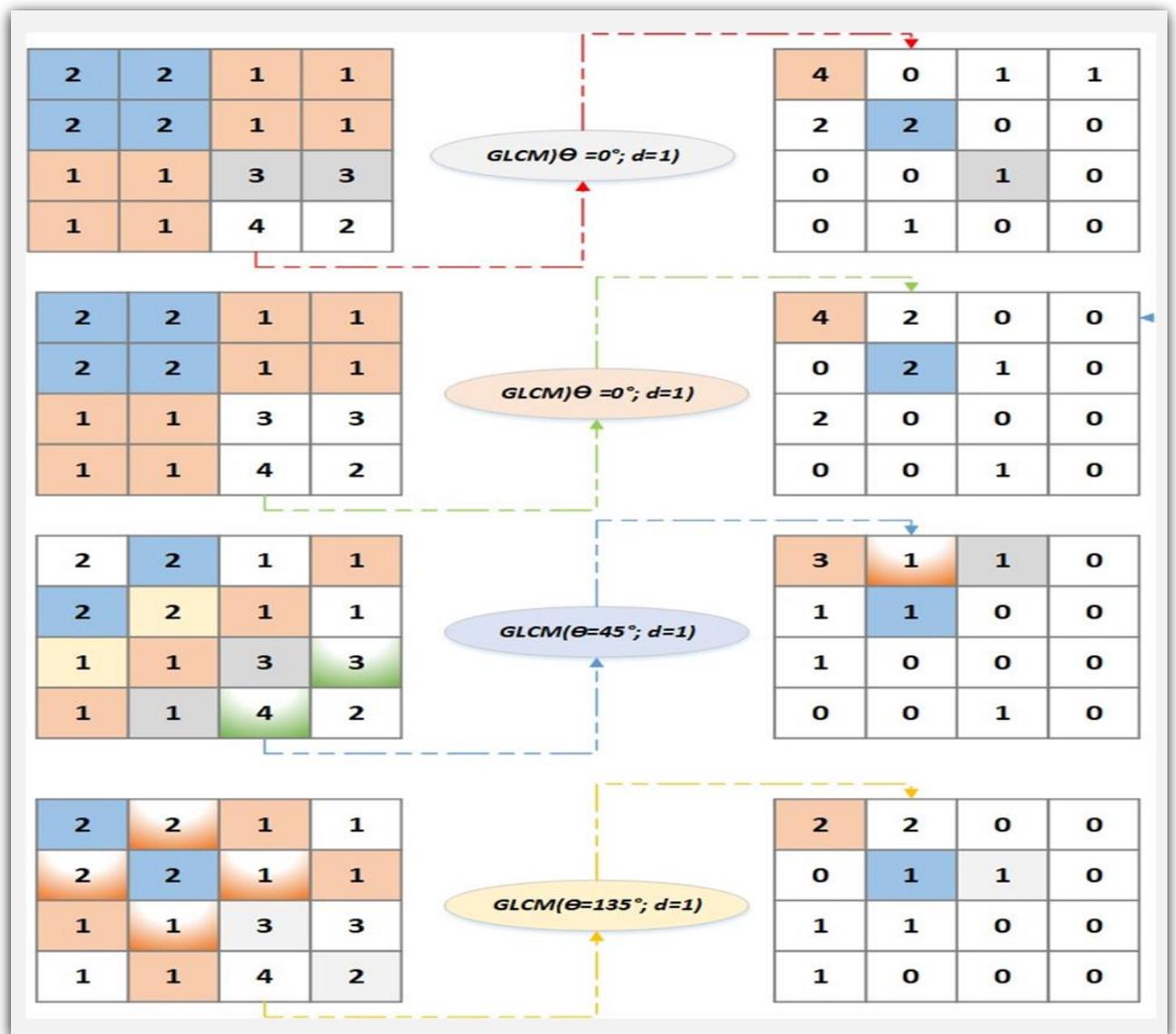


Figure 2- Exemple de matrice de co-occurrence avec $d = 1$ et $\theta = 0^\circ$.

- **Énergie:** Ce paramètre permet de mesurer l'uniformité de la texture. Il atteint de fortes valeurs lorsque les pixels sont similaires en d'autres termes, lorsque la distribution des niveaux de gris est constante ou de forme périodique.

$$P_1 = \sum_{i=0}^{N_{g-1}} \sum_{j=0}^{N_{g-1}} (p(i,j))^2 \quad 2.2$$

(i, j) désigne les coordonnées d'un élément de la matrice P (i, j) désigne la valeur normalisée (divisée par le nb de pixels)

- **Contraste :** Le contraste est une mesure de l'intensité ou des variations du niveau de gris entre le pixel de référence et son voisin. Dans la perception visuelle du monde réel, le contraste est déterminé par la différence de couleur et de luminosité de l'objet et d'autres objets dans le même champ de vision.

$$P_2 = \sum_{i=0}^{N_{g-1}} \sum_{j=0}^{N_{g-1}} ((i,j)^2 \times p(i,j)) \quad 2.3$$

- **L'entropie :** Ce paramètre mesure le désordre dans l'image. Contrairement à l'énergie, l'entropie atteint de fortes valeurs lorsque la texture est complètement aléatoire (sans structure apparente). Elle est fortement corrélée (par l'inverse) à l'énergie.

$$P_3 = \sum_{i=0}^{N_{g-1}} \sum_{j=0}^{N_{g-1}} (\log(P(i,j)))^2 (i,j)$$

2.4

- **La variance** : La variance mesure l'hétérogénéité de la texture. Elle augmente lorsque les niveaux de gris diffèrent de leur moyenne. La variance est indépendante du contraste.

$$P_4 = \sum_{i=0}^{N_{g-1}} \sum_{j=0}^{N_{g-1}} ((i,u)^2 \times p(i,j))$$

2.5

- **La corrélation** : La fonction de corrélation montre la dépendance linéaire des valeurs de niveau de gris dans la matrice de cooccurrence.

$$P_5 = \sum_{i=0}^{N_{g-1}} \sum_{j=0}^{N_{g-1}} P(i,j) \frac{(i,u)(j-u)}{\sigma}$$

2.6

- **Le moment inverse** : (Inverse Difference Moment (IDM)) mesure l'homogénéité de l'image. Ce paramètre est corrélé à une combinaison linéaire des variables d'énergie et de contraste.

$$P_6 = \sum_{i=0}^{N_{g-1}} \sum_{j=0}^{N_{g-1}} P(i,j) \frac{1}{1 + (i,j)}$$

2.7

2.3.2 Motifs binaires locaux (LBP)

Les descripteurs de textures à base des motifs binaires locaux ont été initialement proposés par (Ojala, Pietikäinen) en 1996 [19]. Le calcul de l'image résultante de l'application des motifs binaires locaux s'appuie sur une opération de convolution lors de l'application d'un filtre sur une image numérique. Il suffit de traiter chaque pixel de l'image en considérant les huit pixels de son voisinage immédiat (voisinage N=8). Le voisinage d'un pixel forme une matrice de 3 par 3 pixels où le pixel à traiter est au centre et son voisinage est autour. **La figure 3** représente un exemple d'exécution de l'algorithme d'extraction des motifs binaires locaux relatif aux étapes présentées ci-dessous.

- **Étape 1** : Extraction du voisinage du pixel à traiter : Les huit valeurs de niveau de gris du voisinage du pixel à traiter sont extraites pour former une matrice de 3x3 pixels. Dans cet exemple, chaque pixel possède une valeur de niveau de gris différente. Le pixel en cours de traitement possède la valeur de niveau de gris 40.

- **Étape 2** : Un seuillage est effectué sur la valeur de niveau de gris des pixels du voisinage. Tout pixel possédant une valeur de niveau de gris supérieure ou égale à la valeur de niveau de gris du pixel en cours de traitement est attribué à la valeur 1. La valeur 0 est attribuée à toute valeur de niveau de gris inférieure à celle du pixel en cours de traitement.

- **Étape 3** : Une matrice multiplicatrice est mise en mémoire. Cette matrice sera utilisée dans la prochaine étape de l'algorithme pour décrire la forme binaire locale résultante de façon unique.

- **Étape 4** : Une multiplication élément par élément est effectuée entre la matrice résultante du seuillage de l'étape 2 et la matrice multiplicatrice de l'étape 3.

- Étape 5** : La sommation des valeurs de la matrice résultante de l'étape 4 est effectuée, cette somme est rapportée dans l'image de sortie aux coordonnées correspondantes du pixel à traiter dans l'image d'entrée. L'algorithme ré-exécute les étapes 1 à 5 jusqu'à ce que tous les pixels de l'image d'entrée soient traités. Suivant la procédure d'identification des motifs binaires locaux, un histogramme est calculé pour caractériser la fréquence d'apparition des divers motifs. Le nombre calculé pour chaque pixel à l'étape 5 identifie de façon unique un motif de niveau de gris parmi les motifs possibles. La forme de l'histogramme résultant est caractéristique de la texture étudiée par l'algorithme des motifs binaires locaux.

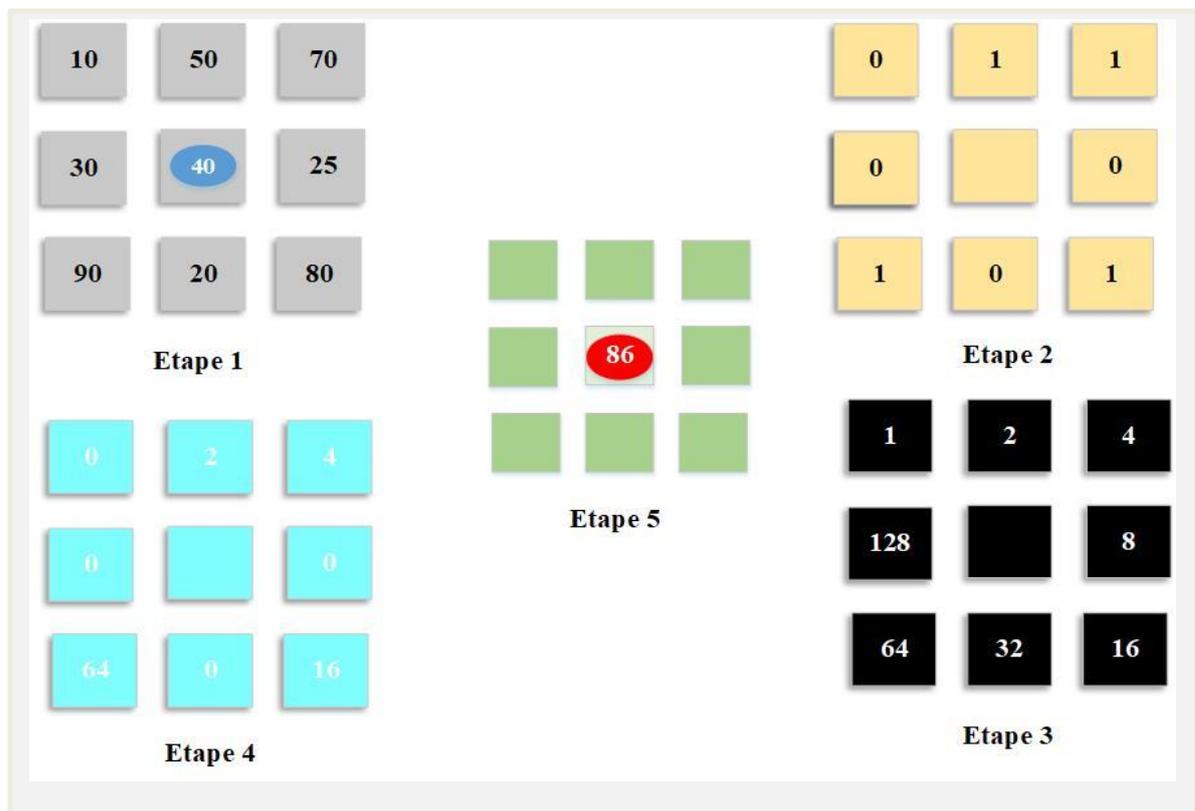


Figure 3 -Le code LBP original.

Pour calculer un code LBP dans un voisinage de P pixels, dans un rayon R , on compte simplement les occurrences de niveaux de gris p plus grands ou égaux la valeur centrale.

$$LBP_{P,R}(X_c, Y_c) \equiv \sum_{p=0}^{P-1} (g_c - g_p) \times 2^p \text{ avec } S(x) \equiv \begin{cases} 1 & x \geq 0 \\ 0 & \text{else} \end{cases} \quad 2.8$$

Où g_i et g_c sont respectivement les niveaux de gris d'un pixel voisin et du pixel central.

(Ojala et al) ont proposé deux variantes de la méthode LBP : la première est définie pour des voisinages de différentes tailles, ce qui permet de traiter la texture à différentes échelles, tandis que la seconde définit ce que l'on appelle les LBP uniformes [20].

- **LBP multi échelle** : Le concept du LBP multi-échelle, est fondé sur le choix du voisinage afin de calculer un code LBP pour pouvoir traiter les textures à différentes échelles. Un voisinage pour un pixel central est réparti sur un cercle et construit à partir de deux paramètres : le nombre de voisins " P " sur le cercle et un rayon " R " pour définir une distance entre un pixel central et ses voisins. Soit une texture $T = t(g_c, g_0, g_{P-1})$, g_c : correspond à la valeur de niveau de gris du pixel central et g_p , avec $P = 0, P - 1$, correspond au niveau de gris de P pixels espacés régulièrement sur un cercle de rayon R . Si les coordonnées de g_c sont égales à $(0,0)$, alors les coordonnées de g_p sont données par l'équation suivante :

$$\begin{cases} x_g \equiv x_c + R \cdot \cos(2\pi \cdot \frac{p}{P}) \\ y_g \equiv x_c + R \cdot \cos(2\pi \cdot \frac{p}{P}) \end{cases}$$

La figure suivante présente des exemples de l'opération LBP ou la notation (P, R) définie les voisinages de centre de cercle P et de rayon R.

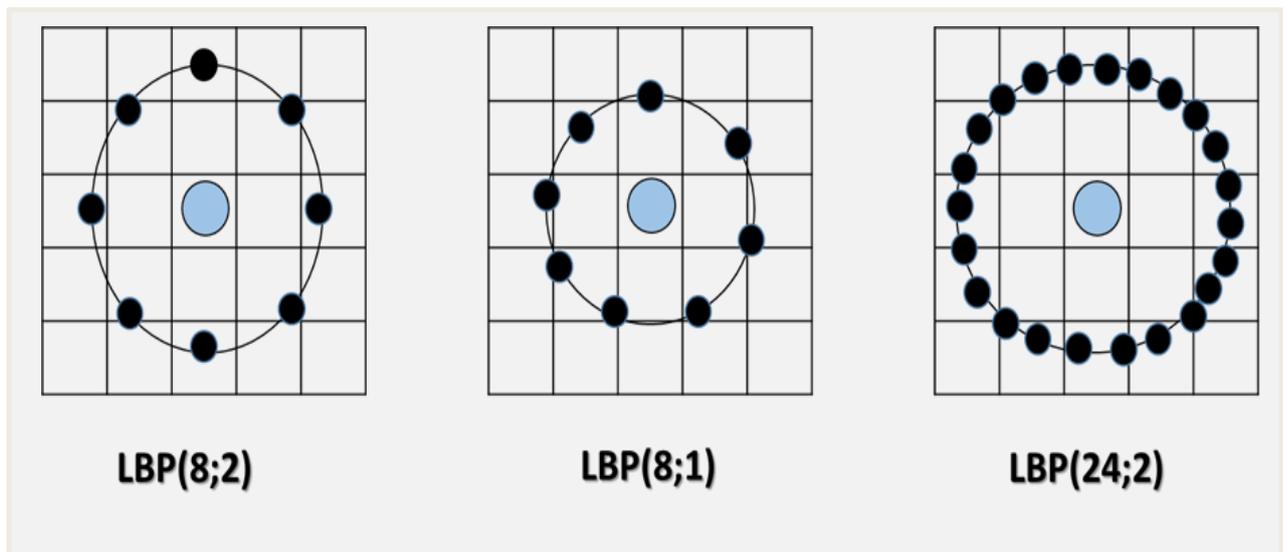


Figure 4-LBP multi-échelle.

Partant de la définition de voisinage, tout d'abord un motif binaire local invariant à toute transformation monotone de l'échelle des niveaux de gris, $LBP_{P,R}$. Pour chaque pixel (x, y) ($g_c = g(x, y)$). Comme dans la méthode LBP classique, le pixel central n'est pas utilisé pour la

caractérisation des textures. En effet, indépendamment du voisinage g_p ce pixel décrit uniquement une intensité lumineuse ce qui n'est pas forcément utile. Par la suite, g_c est utilisé comme un seuil de la manière suivante :

$$T = t(u(g_0 - g_c) \dots \dots \dots, u(g_0 - g_{p-1}))$$

2.10

Finalement, Le calcul du code LBP est obtenu de la même manière que le LBP basique.

- **LBP invariant par rotation**

Quant à la seconde méthode dite, LBP invariant par rotation ou motif binaire uniforme, est basée sur la notion d'uniformité. Cette notion permet de représenter les informations de primitives Structurelles comme les arêtes (coins) et les contours.

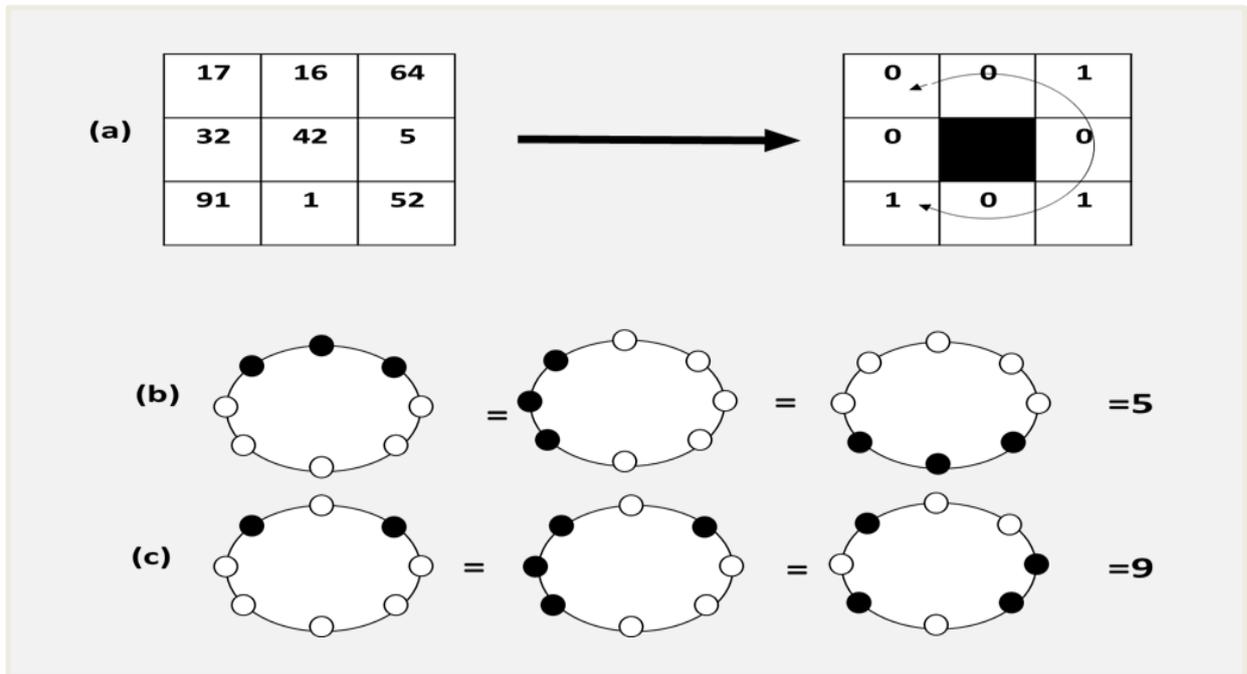


Figure 5-Construction et uniformité d'un motif LBP

Pour tout pixel (x, y) le calcul du motif est donné par (Ojala) :

$$(LBP_{P,R}^{ri})(x, y) = \min\{ROR(LBP_{P,R}(x, y)), i/i \in [0 \dots p-1]\} \quad 2.11$$

(ROR) (a, i) correspond au résultat de i décalages circulaires successifs vers la droite des bits du nombre a (codé sur P bits). $LBP_{P,R}^{ri}$ Quantifie ainsi les statistiques d'occurrence de modèles individuels invariant en rotation correspondant à certaines micro-caractéristiques de l'image. Pour une valeur $R = 1$, $LBP_{P,R}^{riu2}$ Est l'équivalent de LBP_{ROT} proposé par (Pietikainen et al)

qui est moins performant. Pour remédier ce problème, une mesure est effectuée sur les motifs pour déterminer leur uniformité. L'opérateur est appelé $LBP_{P,R}^{riu2}$ et défini de la manière suivante :

$$LBP_{P,R}^{riu2} = \begin{cases} \sum_p^{p-1} U(g_0 - g_c) & \text{si } U(LBP_{P,R}) \leq 2 \\ P + 1 & \text{sinon } 0 \end{cases} \quad \text{2.12}$$

Concernant les motifs uniformes, l'opérateur 2 LBP correspond à la somme des bits du voisinage. Au contraire si les motifs sont non-uniformes, l'opérateur est affecté à la valeur $1 + p$. Le nombre d'apparitions de chaque code est accumulé dans un histogramme de taille $2 + p$. Chaque tranche de cet histogramme donne une estimation de la probabilité d'apparition d'un motif dans une texture. Les textures sont caractérisées en utilisant directement l'histogramme des codes binaires. Dans [21], les auteurs proposent une variation de cette méthode en coordonnant une invariance à la rotation à la fois globale et locale. Dans un premier temps, ils utilisent la distribution LBP pour déterminer l'orientation principale d'une texture et l'utilisent afin d'aligner l'histogramme LBP. Le motif uniforme obtient le même code LBP quelle que soit son orientation. Le fait d'aligner l'histogramme LBP selon l'orientation principale (information globale) permet de différencier ces 3 motifs uniformes en leur affectant des codes binaires différents. Enfin, les auteurs proposent également un nouveau descripteur de textures LBP variance (LBPV) permettant de mesurer les informations locales de contraste.

- LBP pour la reconnaissance de la maladie de parkinson :** Une fois le code LBP calculé pour tous les pixels de l'image, on calcule l'histogramme de cette image LBP pour former un vecteur de caractéristiques représentant l'image spirale. En réalité, afin d'incorporer plus informations spatiales au vecteur représentant l'image spirale dessiner par le patient de la maladie de parkinson, on divise tout d'abord l'image codée par l'opérateur LBP en petites régions et l'histogramme est construit pour chaque région.

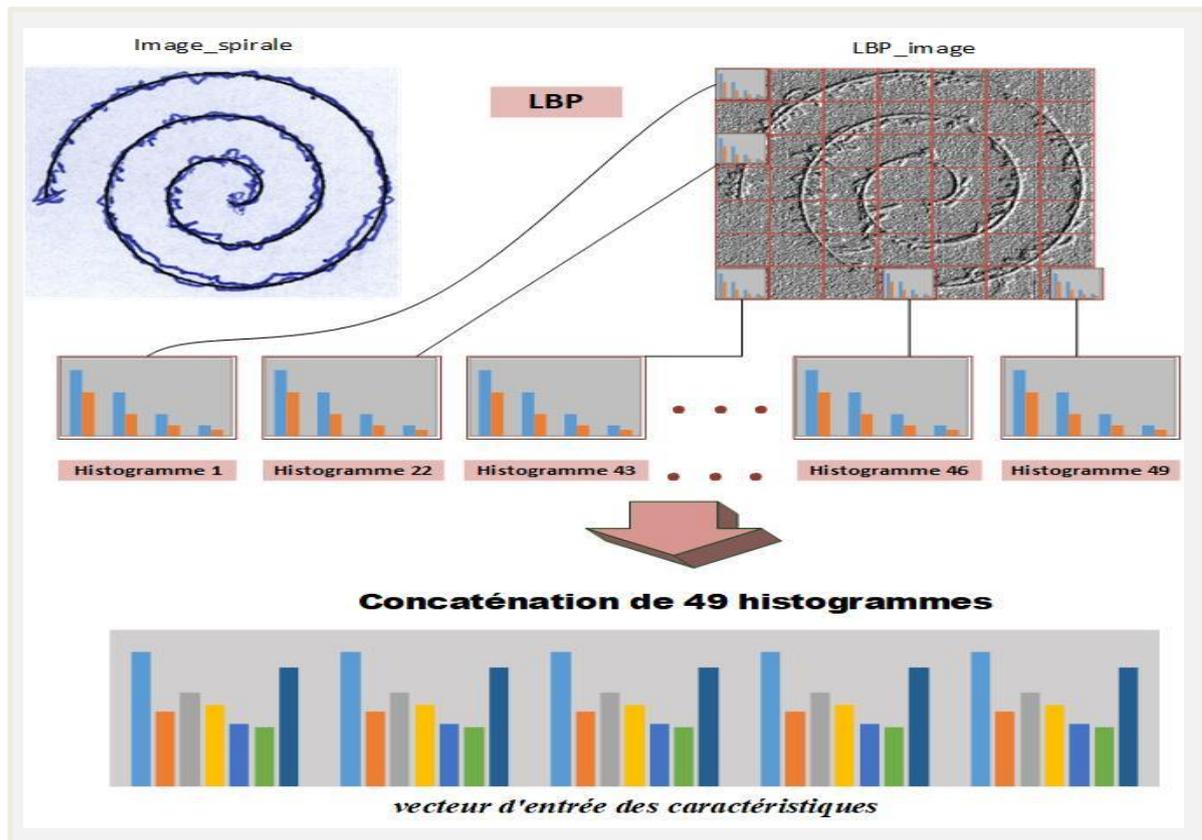


Figure 6-Représentation d'une image spirale par les histogrammes du code LBP.

2.3.3 Histogramme à base du gradient orienté (HOG)

HOG est un descripteur très puissant proposé par (Dalal et al) en 2005, qui a été initialement développé pour la détection humaine, mais plus tard il est étendu et appliqué à de nombreux autres problèmes de vision par ordinateur y compris la reconnaissance de la maladie. HOG décrit l'aspect et la forme locale de l'objet dans une image en utilisant la distribution des gradients. Le vecteur caractéristique d'une image (x, y) par la technique HOG est obtenu par la procédure suivante :

- **Étape 1** : Diviser l'image (x, y) en N cellules régulières (N petites régions) et des blocs. Les valeurs de gradient sont calculées pour chaque pixel en utilisant un filtre dérivatif 1-D centré, dans les directions horizontales et verticales. Pour cela les masques suivants sont utilisés :

$$D_x = [-1 \ 0 \ +1] \quad \text{2.13}$$

$$D_y = \begin{bmatrix} -1 \\ 0 \\ +1 \end{bmatrix} \quad \text{2.14}$$

- **Étape 2** : L'amplitude et l'orientation de gradient à chaque pixel (x, y) sont calculées en utilisant les équations.

$$|G(x, y)| = \sqrt{G_x^2(x, y) + G_y^2(x, y)}$$

2.15

$G_x(x, y)$ et $G_y(x, y)$ correspondent au gradient horizontale et au gradient verticale, au pixel (x, y) respectivement. Pour construire le vecteur de descripteur de l'histogramme du gradient orienté, chaque image est subdivisée en cellules homogènes. Les histogrammes de contour sont agrégés partialement pour chaque cellule. L'agrégation spatiale offre une robustesse vis-à-vis des petites variations de forme et réduit ainsi la taille du vecteur du descripteur, ceci rend le descripteur HOG très simple et surtout efficace pour les systèmes temps réels. L'idée de base alors est de projeter chaque pixel dans chaque cellule. Ensuite pour chaque cellule, on construit un histogramme et on concatène tous les histogrammes partiels dans un histogramme global HOG.

• **Étape 3 :** L'histogramme de l'orientation du gradient à l'intérieur de chaque cellule est calculé par quantification Gradients non signés à chaque pixel en 9 canaux orientations. Les histogrammes sont uniformes de 0 à 180° (cas non signé) ou de 0 à 360° (cas signé).

$$\theta(x, y) = \arctg\left(\frac{G_y(x, y)}{G_x(x, y)}\right)$$

2.16

• **Étape 4 :** Le vecteur caractéristique pour chaque cellule est normalisé en utilisant des vecteurs dans leurs blocs respectifs. Dans ce travail nous utilisons la méthode de la norme L2- pour la normalisation des blocs, le facteur de normalisation est calculé à l'aide de l'équation (2.17).

$$V_n = \frac{v}{\sqrt{\|v\|_2^2}} + \epsilon$$

2.17

v est le vecteur non normalisé contenant tous les histogrammes dans un bloc, $\|v\|_2$ est la norme- L2 du vecteur descripteur, et ϵ est un terme de régularisation.

Il existe d'autres facteurs de normalisation comme la norme L1 qui est définie par :

$$V_n = \frac{v}{\|v\|_1} + \epsilon$$

2.18

- **Étape 5** : Le vecteur caractéristique de chaque bloc (c) est formé en concaténant les vecteurs d'histogramme de toutes les cellules dans le bloc. À la fin le vecteur caractéristique HOG est formé en concaténant les vecteurs de caractéristiques de tous les blocs pour une image donnée.

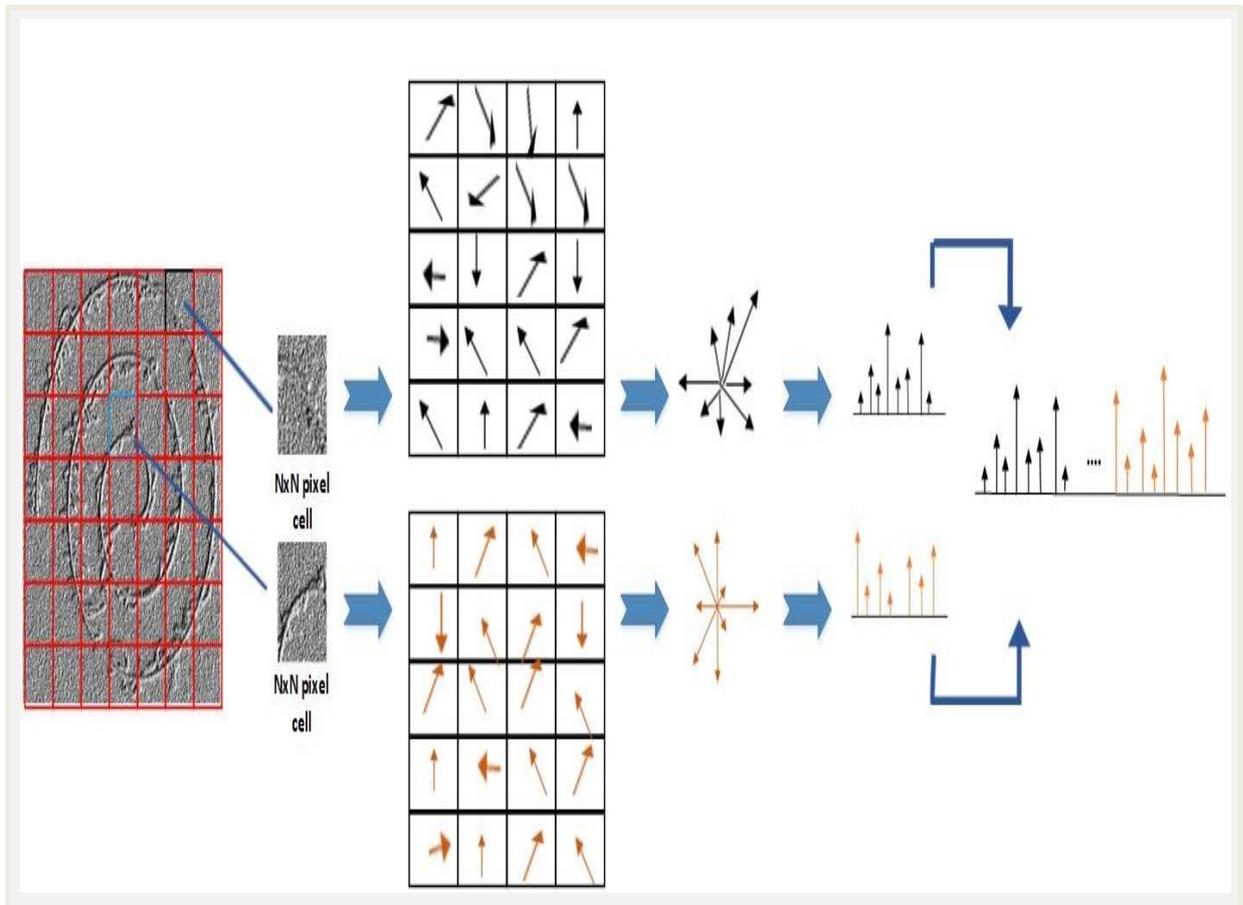


Figure 7-Principe de la décomposition pour le calcul des HOG

2.4 Méthodes de classification

2.4.1 Le modèle de Perceptron Multi-Couches (MLP)

Le Perceptron Multi Couches (PMC), en anglais Multi-Layers Perceptron (MLP), est le réseau de neurones le plus utilisé pour des fonctions d'approximation, de diagnostic et de

prédiction. Les réseaux de neurones non bouclés (Feed-Forward Neural Networks) représentent une catégorie de réseaux de neurones artificiels dont les nœuds forment un graphe orienté acyclique dans lequel l'information circule dans un seul sens, c'est-à-dire, de l'entrée vers la sortie. Les réseaux multicouches présentent l'intérêt de posséder un algorithme d'apprentissage supervisé bien adapté, appelé l'algorithme de rétro-propagation du gradient. La fonction de transfert souvent utilisée dans les PMCs est une fonction non-linéaire, sigmoïde qui permet un apprentissage plus rapide.

2.4.1.1 Architecture

Un MLP est constitué de trois types de couche, une couche d'entrée, une couche de sortie et une ou plusieurs couches cachées :

- ***Une couche d'entrée est la première couche du réseau.*** Les activations de cette couche réceptionnent l'information fournie par les vecteurs d'entrée de chaque instance. Cette couche ne comporte donc pas de connexions en entrée venant d'autres nœuds. Elle est par contre complètement connectée à la première couche cachée.
- ***Les couches cachées ou couches de prétraitement.*** Ces couches sont placées entre les entrées et la couche de sortie. Leur rôle est de préparer les données en utilisant principalement des fonctions d'activation non-linéaires dans leurs neurones pour les présenter à la couche de sortie. Pour cela, ces couches doivent être de type perceptron simple et non pas de type ADALINE. Car, en combinant plusieurs fonctions linéaires on obtient une fonction linéaire. Le nombre de neurones sur chaque couche cachée ainsi que le nombre de couches cachées dépendent du traitement à effectuer ainsi que du comportement du réseau souhaité.

- ***Une couche de sortie appelée souvent couche de décision.*** C'est la couche à l'extrémité droite. Sur cette couche, les résultats finaux de traitement seront récupérés. Le nombre de neurones sur la couche de sortie dépend donc des résultats désirés. Cette couche peut être présentée par un perceptron simple. Entre autres, elle peut être présentée par un autre modèle neuronal appelé ADALINE. Ce modèle à la même architecture que le perceptron simple sauf que les neurones utilisés par ce modèle, ont une fonction d'activation linéaire. **La figure 8** montre la composition générique d'un perceptron multi couche. Dans ce type de réseaux, les neurones sont organisés en trois niveaux ou plus appelés « couches ». Les couches voisines sont complètement connectées, c'est-à-dire, les nœuds de chaque couche sont liés à tous les nœuds de la couche inférieure et à tous ceux de la couche supérieure. En revanche, aucune connexion n'existe entre les unités d'une même couche.

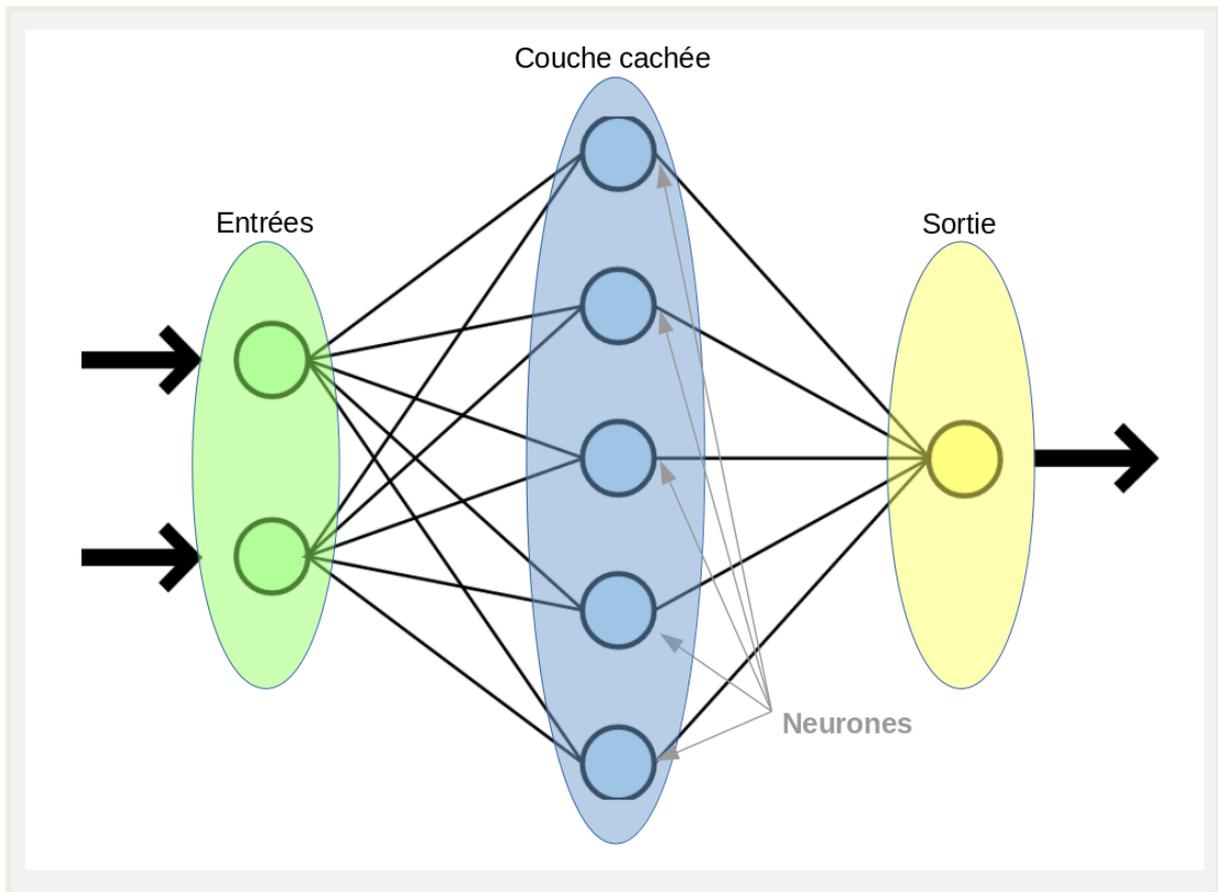


Figure 8-Exemple d'un perceptron multicouche

2.4.1.2 Rétro-propagation du gradient

La phase d'apprentissage des MLP consiste à adapter les poids des connexions en fonction des erreurs de prédiction constatées à chaque classification d'une nouvelle instance. La rétro propagation du gradient (back-propagation) : est la méthode la plus utilisée pour l'adaptation des poids. Cet algorithme permet de déterminer le gradient de l'erreur pour chaque

neurone du réseau en partant de la dernière couche et en arrivant jusqu'à la première couche cachée. L'objectif de la rétro-propagation du gradient est d'ajuster les poids des connexions dans le but de minimiser l'erreur quadratique qui représente l'écart entre la sortie attendue et la sortie produite par le réseau (hypothèse) correspondant à un vecteur d'entrée donné. N représente la taille des vecteurs en sortie.

$$E = \frac{1}{2} \sum_{i=1}^N (ref_i - hyp_i)^2 \quad 2.19$$

La rétro-propagation du gradient est considérée comme un problème d'optimisation pour lequel nous pouvons utiliser la méthode de descente de gradient. Considérant que l'erreur E est fonction des poids w , Un minimum local est visé en changeant les poids dans le sens inverse du gradient $\frac{\delta E}{\delta W}$ multiplié par le taux d'apprentissage α :

$$\Delta w *_{ij} = -\alpha \frac{\partial E}{\partial w_{ij}} \quad 2.20$$

$$w_{ij} = w_{ij} + \Delta w *_{ij} \quad 2.21$$

Prenons un réseau de neurones MLP contenant L couches. La couche ($2 \leq K \leq L$) contient N_k neurones à laquelle est affecté un vecteur de biais b^k et une matrice de poids w . Dans cette matrice, un élément w_{ij}^k représente le poids de la connexion partant du neurone i ($1 \leq i \leq N_{k-1}$) de la couche $k-1$ vers le neurone j ($1 \leq j \leq N_k$) de la couche k . Pour chaque nouvelle instance, une passe d'apprentissage est effectuée en 3 étapes :

1. **Propagation vers l'avant** : Le vecteur d'entrée est copié dans les activations a_i^l de la première couche. Ensuite, pour chacune des k couches, partant de la première couche cachée et montant jusqu'à la couche de sortie, le potentiel biaisé p_i^k et l'activation a_i^k du neurone i sont calculés.
2. **Propagation vers l'arrière** : Calculer la dérivée Δ_i^L de l'erreur E pour l'activation a_i^L du neurone i de la couche de sortie :

$$\Delta_i^L = (ref_i - a_i^L) \frac{\partial f}{\partial p_i^L}$$

2.21

Ensuite, en descendant à partir de la dernière couche cachée $h=L-1$ jusqu'à la première $h=2$, calculer le terme Δ_i^L pour chaque neurone :

$$\Delta_i^h = \sum_{j=1}^{U_{h-1}} \Delta_j^{h+1} w_{ij}^h \frac{\partial f}{\partial w_{ij}^h}$$

2.22

3. Adaptation des paramètres. Mettre à jour le biais et les poids des nœuds j pour chaque couche k :

$$b_j^{k*} = b_j^k + \alpha \Delta_i^k$$

2.23

$$w_{ij}^{k*} = w_{ij}^k + \alpha \Delta_i^k a_j^k$$

2.24

La phase d'apprentissage est beaucoup plus coûteuse que celle de classification. Cette dernière consiste simplement, à propager vers l'avant, à l'instar de l'étape 1 de l'algorithme d'apprentissage, les données de chaque nouvelle instance afin d'obtenir le résultat de classification automatique.

2.4.1.3 Avantages du perceptron à une couche cachée

Parmi les nombreux modèles non linéaires neuronaux qui peuvent être utilisés pour estimer une relation non linéaire, le perceptron à une couche cachée qui présente des caractéristiques intéressantes, que l'on peut résumer comme suit : C'est un modèle boîte noire non linéaire, à capacité d'approximation universelle, flexible et parcimonieux.

- **Boîte noire non linéaire** : comme leur nom l'indique, les PMCs sont organisés autour d'un ensemble de cellules ou neurones arrangés dans des couches. Il est considéré comme une boîte noire qui reçoit des entrées et émet une sortie. Capacité

d'approximation universelle : plusieurs travaux ont montré que le perceptron à une couche cachée est un approximateur universel, c'est-à-dire qu'il est capable d'approximer n'importe quelle fonction de transfert si le nombre d'unités cachées n'est pas grand.

- **Flexibles** : plus la relation (non linéaire) à modéliser est complexe, plus les neurones et les paramètres dans le modèle neuronal correspondant seront nombreux. Cela signifie que leur complexité interne peut être facilement accrue, sans changer la forme globale du modèle
- **Parcimonieux** : les perceptrons à une couche cachée, qui sont des développements en série de fonctions paramétriques, c'est-à-dire impliquant des paramètres ajustables à estimer, nécessitent moins de paramètres que les développements en série de fonctions fixes pour atteindre un même degré d'approximation. En ce sens, ils sont parcimonieux.

2.4.2 Méthode des k plus proches voisins (KNN)

L'algorithme des k plus proches voisins (noté k - PPV) est une famille de techniques pouvant être utilisées pour la classification ou la régression. En tant qu'un algorithme d'apprentissage non-paramétrique, les k -plus proches voisins ne sont pas limités à un Nombre déterminé de paramètres. Par contre, leur complexité dépendra de la taille de la base d'apprentissage. Le principe de la méthode k -NN est de trouver k plus proches voisins, à partir de l'échantillon d'apprentissage, à une nouvelle instance qu'on cherche à classer. La classe de la nouvelle instance est la classe majoritaire (la plus représentée) parmi ces k voisins. Dans le cas d'une régression, la valeur de sortie est une valeur continue qui peut être, par exemple, la moyenne des valeurs des k voisins. Cette méthode dépend donc des trois éléments suivants :

- L'échantillon d'apprentissage.
- La mesure de similarité (ou de distance).
- La méthode de combinaison des voisins.

Cette méthode utilise principalement deux paramètres : une fonction de similarité pour comparer les individus dans l'espace de caractéristiques et le nombre k qui décide combien de voisins influencent la classification. Pour tester la similarité entre deux vecteurs, Il existe plusieurs fonctions pour calculer la distance parmi lesquels on trouve :

- **La distance Euclidienne** : La distance euclidienne est la plus utilisée, elle est définie comme suit :

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$
2.25

Où : x, y sont des vecteurs.

- **La distance de Manhattan :**

$$d(x, y) = \sum_{i=1}^n |x_i - y_j|$$

2.26

Où : x, y sont des vecteurs.

- **La distance de Minkowski (p-distance) :** La distance de Minkowski est une généralisation de la distance Euclidienne et la distance de Manhattan. Elle est définie comme suit :

$$d(x, y) = (\sum_{i=1}^n |x_i - y_j|^p)^{1/p}$$

2.27

Où : x, y sont des vecteurs, p : paramètre de contrôle.

Il est important de noter que, KNN ne nécessite pas une phase d'apprentissage ou un modèle. Les exemples d'apprentissage sont des vecteurs dans un espace multidimensionnel avec un label de classe d'appartenance, ils sont stockés en permanence dans la mémoire lors de la phase de classification.

2.4.3 Méthodes d'apprentissage machine classique

2.4.3.1 Machine à vecteurs de support (SVM)

La classification avec la SVM repose sur la détermination d'un hyperplan qui sépare deux classes. L'hyperplan est défini en fonction de la marge maximale. La marge maximale est la distance entre l'hyperplan et les échantillons les plus proches de chaque classe. Toutefois, afin de ne pas se limiter aux données linéairement séparables, il est possible d'utiliser un noyau afin de séparer les données non linéairement séparables. Cela consiste à transformer la représentation des données dans un espace plus grand où elles seront linéairement séparables. Pour se faire, on utilise le produit scalaire de nos données avec l'espace de plus grandes dimensions. Les noyaux non linéaires permettent de faire la correspondance avec cet espace de plus hautes dimensions. Parmi les types de noyaux communément utilisés, on retrouve la fonction de base radiale (RBF), la fonction polynomiale, et la fonction gaussienne **figure 9**.

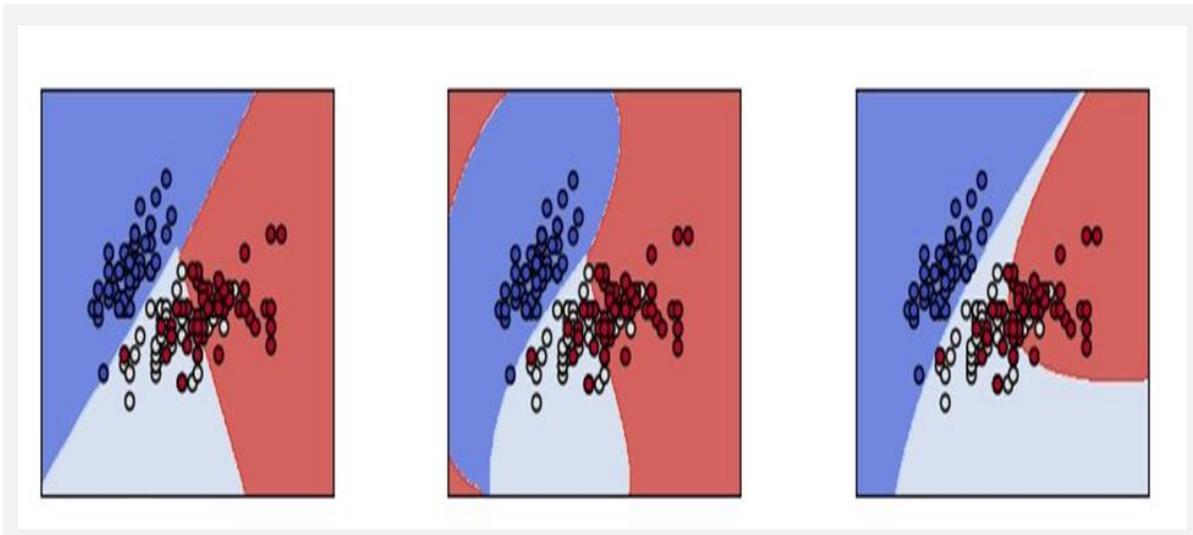


Figure 9- La classification avec la SVM.

La régularisation du SVM est une étape importante pour optimiser la classification. Un des facteurs de régularisation les plus importants est le paramètre C. Celui-ci est impliqué dans la fonction de minimisation des poids de l'hyperplan de la manière suivante :

$$\min \left(\frac{1}{2} \|W\|^2 + C \sum_{i=1}^m \varphi \right) \quad 2.28$$

où W correspond aux poids (coefficients) de l'hyperplan, et φ est une variable proportionnelle aux erreurs de classification. Ainsi, le paramètre C indique le coût $C \sum_{i=1}^m \varphi$ lorsque l'algorithme fait des erreurs. C'est un indicateur du compromis qu'on est prêt à faire pour maximiser la marge de l'hyperplan vis-à-vis la présence d'erreurs de classification. Par conséquent, des grandes valeurs de C vont faire pencher l'algorithme à choisir des marges petites, si ces marges permettent de bien classifier les données d'entraînement. Toutefois, ceci pose un risque de sur-apprentissage. De l'autre côté, de faibles valeurs de C vont plutôt tendre l'algorithme à maximiser sa marge, même si cette marge diminue la précision de la classification **figure 10**.

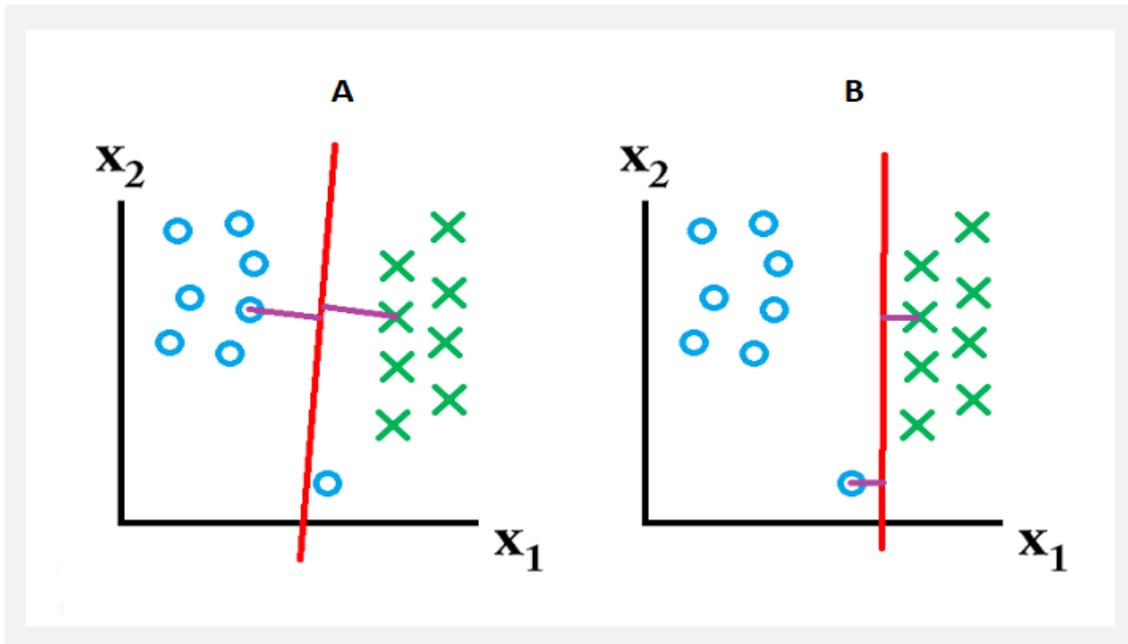


Figure 10 - Effet du coefficient de régularisation

2.5 Conclusion

Dans ce chapitre, on a présenté les différentes classes des méthodes d'analyse de texture. Puis, on a introduit les trois types de classification, le premier est un classifieur à base de distance tandis que le second est un classifieur neuronal et le troisième SVM. Dans le prochain chapitre on va introduire les algorithmes d'optimisation avancés inspirés de la nature et de la physique.

Chapitre 3

Les techniques d'optimisation avancées

3.1 Introduction

Les algorithmes évolutionnaires sont des méthodes de recherche efficaces pour résoudre des problèmes complexes en sciences, en optimisation et en industrie. Même si ces heuristiques permettent de réduire de manière significative le temps de calcul de l'exploration de l'espace de recherche d'une solution, elles souffrent d'un problème important qui se pose sous le thème d'ajustement et contrôle de paramètres de l'algorithme. Ce chapitre s'attache à présenter en détail les algorithmes évolutionnaires, le principe de l'évolution et tout ce qui est en rapport et sert à bien définir cette grande famille de méthodes d'optimisation, parmi les algorithmes d'optimisation (HHO, GWO, AOA, MPA) prise de la bibliothèque (Mealpy).

3.2 Algorithmes évolutionnaires

Les méthodes évolutionnaires représentent une classe de méta-heuristique ayant comme principe de faire évoluer une population de base (constituée de solutions au problème à résoudre) par des opérateurs de variations (croisement et mutation). Cette population est soumise à une sélection pour le croisement puis une sélection pour la survie. L'objectif est de faire converger les solutions (appelées « individus ») vers un optimum en respectant certains critères.

3.2.1 Principe des algorithmes évolutionnaires

Un algorithme évolutionnaire peut être vu comme un processus dynamique qui fait évoluer une population de solutions potentielles, ainsi qu'un instrument de mesure qui permet d'évaluer leurs valeurs : la fonction objectif ou fitness. Une boucle dans l'algorithme ou génération simule l'évolution à travers le temps, pendant laquelle quatre étapes se succéderont : l'évaluation, la sélection, l'évolution et le remplacement [22].

Soit F une fonction à optimiser, définie sur l'espace de recherche Ω ; nous l'appellerons la fonction de fitness. Les points de l'espace Ω notés par des vecteurs, X_i Appelés individus et la population notée par π est un ensemble d'individus, que nous considérons par la suite de taille p ; i.e.

$\pi = \{X_1, X_2, X_3, \dots, X_p\}$ L'algorithme évolutionnaire a la structure suivante :

- Initialisation : la population initiale π_0 est générée en tirant aléatoirement p individus dans Ω .
- Boucle des générations indexées par t .
- **Évaluation** : pour chaque individu, calculer sa valeur en termes de la fonction de fitness i.e. (la valeur de $F(X_i)$).

Algorithme1 : Procédure d'un algorithme évolutionnaire

1. Début
 2. $T \leftarrow 0$
 3. Initialiser π_0
 4. Évaluer π_0
 5. Tant que (la condition n'est pas terminée) faire
 6. $T \leftarrow t + 1$
 7. Sélectionner π_t de π_{t-1}
 8. Appliquer les opérateurs de croisement et de mutation π_t
 9. Evaluer π_t
 10. Fin
 11. Fin
-

- **Sélection** : La sélection parmi les individus de π_{t-1} des géniteurs.
- **Evolution** : Les opérateurs de variation sont appliqués aux individus géniteurs pour générer les enfants.
- **Remplacement** : Les nouveaux individus (enfants) sont introduits dans π_{t-1} pour créer une nouvelle population.
- **Arrêt** : Si le critère d'arrêt est atteint, par exemple si la performance souhaitée est atteinte, s'il n'y a plus d'améliorations de F ou si le temps de calcul permis est écoulé.

3.2.2 Mise en œuvre des algorithmes évolutionnaires

La mise en œuvre d'un algorithme évolutionnaire nécessite plusieurs étapes de conception et de choix de certains paramètres. En effet, il est important de choisir le codage des solutions, l'opérateur de sélection et le processus de renouvellement de la population, via les opérateurs de croisement et de mutation.

3.2.2.1 Le codage ou représentation

Il existe essentiellement deux types de codage : le codage binaire ou représentation sous forme de chaîne binaire et le codage réel qui est une représentation directe des valeurs réelles de la variable. Nous pouvons naturellement passer d'un codage à l'autre. Le codage initialement utilisé par John Holland dans l'AG standard est le codage binaire. Nous allons par conséquent d'abord présenter ce cas [23,24].

- **Représentation binaire** : Par analogie (imitation) avec la génétique naturelle, les AEs utilisent habituellement des chaînes de bits pour représenter les chromosomes. Ainsi, pour un problème d'optimisation ayant n variables entières, on peut représenter chacune de ces variables par un vecteur binaire de taille égale à k bits (suivant leur domaine) ; on obtiendra ainsi un chromosome final de taille $(n*k)$ Les premiers résultats de convergence dans la littérature des AEs ont été basés sur de telles chaînes binaires. Ils ont montré que le codage des chromosomes à l'aide de gènes dont l'alphabet possède un faible cardinal était théoriquement plus efficace.

Le codage binaire confère en outre aux AEs une très grande robustesse car il est indépendant du domaine du problème traité et les opérateurs stochastiques standards peuvent être systématiquement utilisés [25].

- **Représentation réelle** : Le principe de cette représentation consiste à coder directement les variables du problème dans l'individu sans passer par le codage binaire intermédiaire [26,27]. Ainsi, les individus ne sont plus représentés par des chaînes binaires mais par des vecteurs réels. C'est bien sûr le cas le plus fréquent en calcul numérique ; on parle alors aussi d'optimisation paramétrique.

L'un des avantages majeurs de cette représentation est de conserver les variables du problème dans le codage lui-même, ce qui lui permet une meilleure prise en compte de la structure même du problème. Cette représentation directe des paramètres réels nécessite de définir de nouveaux opérateurs de variations génétiques bien spécifiques à ses caractéristiques.

3.3 Les différents types d'algorithmes évolutionnaires

3.3.1 Les algorithmes inspirés de l'intelligence distribuée

La diversité des phénomènes dans la nature a été une riche source d'inspiration pour les scientifiques qui ont proposé un grand nombre d'algorithmes bio-inspirés pour l'optimisation. Ces algorithmes s'inspirent en particulier de l'évolution naturelle comme les algorithmes génétiques et l'évolution différentielle. En plus, un nouveau comportement d'inspiration est introduit récemment à base d'intelligence distribuée (en anglais Swarm Intelligence) qui cherche à imiter le comportement collectif des espèces telles que les insectes et les animaux.

3.3.1.1 L'algorithme des loups gris (GWO)

L'optimiseur de loup gris (GWO) est un algorithme méta-heuristique basé sur la population qui simule la hiérarchie de leadership et le mécanisme de chasse des loups gris dans la nature, et proposé par (Seyedali et al) en 2014. Cette hiérarchie emploie quatre sortes de loups : alpha (α), bêta (β), delta (δ) et oméga (ω). Alpha est le leader, il est chargé de prendre des décisions.

Alpha est le leader, il est chargé de prendre des décisions. Il est également appelé le loup dominant, car il doit être suivi par la meute. C'est le meilleur membre pour gérer le pack, mais ce n'est pas automatiquement le loup le plus fort. Les loups bêta sont au deuxième niveau de la hiérarchie. Ils aident les loups alpha à prendre des décisions mais doivent les respecter et commandent les loups du niveau inférieur. Dans le cas où l'alpha décède, les bêtas sont les candidats appropriés pour le remplacer. Omega est le loup gris le moins bien classé. Cette catégorie de loups doit appliquer toutes les décisions des autres loups dominants. Si un loup n'est pas un alpha, un bêta ou un oméga, il est appelé delta (ou subordonné dans certaines références). Les loups de ce niveau dominent les omégas mais doivent se soumettre aux alphas et aux bêtas. Les loups gris ont un autre comportement social intéressant en plus de la hiérarchie sociale. Ce comportement est la chasse en groupe. Les principales phases de cette technique sont les suivantes : rechercher des proies, les encercler et les attaquer par la suite. [28] L'algorithme d'optimisation Grey Wolf (GWO) est une nouvelle méthode d'optimisation très efficace. L'algorithme GWO peut révéler d'une performance efficace en comparaison avec d'autres algorithmes d'optimisation et largement utilisé au cours de la dernière décennie pour différentes applications.

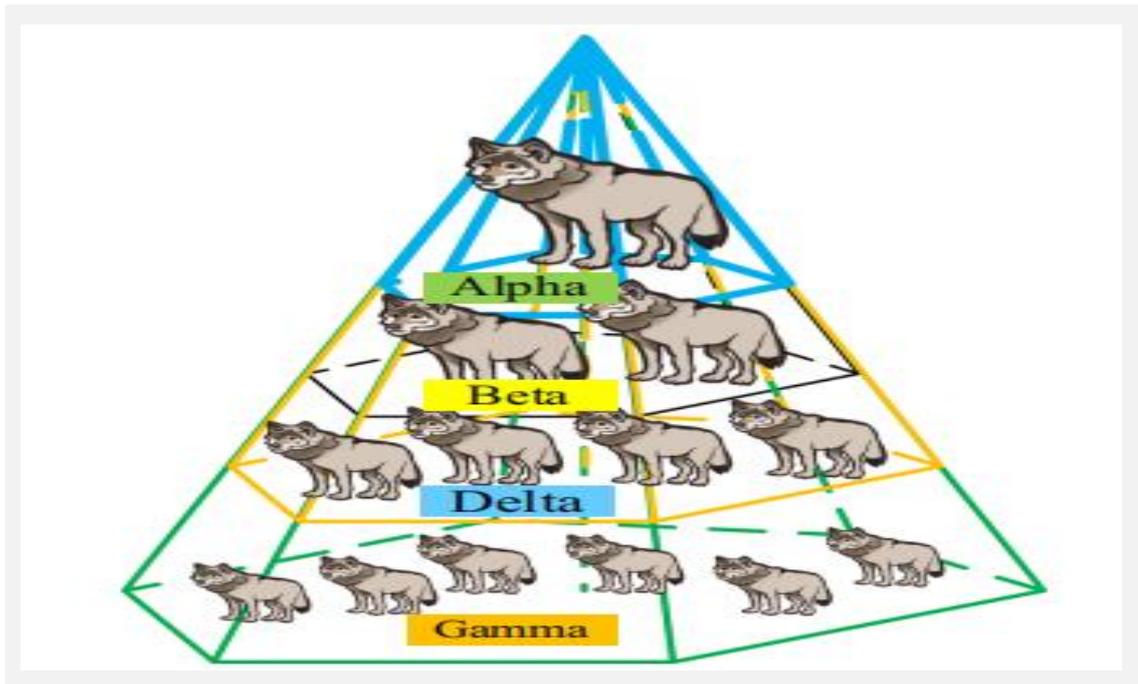


Figure 11- La hiérarchie sociale des loups gris

Le premier élément est considéré comme le mieux adapté à la solution qui est noté alpha (α), suivi par bêta (β) et delta (δ). Le reste des éléments sont considérés comme oméga (ω). Dans l'algorithme GWO, la chasse (optimisation) est contrôlée par les loups du premier ordre, second ordre et du troisième ordre.

Afin de modéliser mathématiquement la hiérarchie sociale des loups lors de la conception de GWO, nous considérons la meilleure solution obtenue par le loup alpha. Par conséquent, la deuxième et la troisième meilleure solution sont nommées: (α), bêta (β), delta (δ)

respectivement. Le reste des solutions candidates sont supposées être des omégas (ω). Dans l'algorithme GWO, la chasse (optimisation) est guidée par les trois catégories des loups.

• **Recherche de proies** : Conformément aux positions des loups alphas, bêtas et deltas, les loups gris recherchent principalement les proies. Ils convergent les uns des autres pour attaquer les proies et divergent pour les rechercher. Afin de modéliser mathématiquement la divergence, nous utilisons le paramètre A avec des valeurs aléatoires supérieures à 1 ou inférieures à 1 pour obliger l'agent de recherche de diverger de la proie. La valeur de ce paramètre est contrôlée par le paramètre a, qui diminue linéairement de 2 à 0. C'est un autre paramètre de GWO qui favorise l'exploration. Il contient des valeurs aléatoires comprises dans l'intervalle [0;2]. Cette contribution est forte lorsque $C < 1$, la solution gravite davantage vers les proies, favorisant l'exploration et évitant les optima locaux.

La proie est encerclée pendant la chasse par des loups gris. Pour modéliser mathématiquement ce comportement, les équations suivantes sont proposées :

$$X(t + 1) = X_p(t) - A \times D$$

3.1

Où t indique l'itération en cours, $X(t + 1)$ est l'emplacement suivant du loup. $X_p(t)$ est un vecteur qui dépend de l'emplacement de la proie, qui est calculé comme suit :

$$D = |C \cdot X_p(t) - (t)|$$

3.2

Les vecteurs A et C sont calculés comme suit :

$$A = 2a \times r_1 - a$$

3.3

$$C = 2 * r_2$$

3.4

Où a est diminué linéairement de 2 à 0 avec $a = 2 - t * \frac{2}{T}$ tandis que r_1 et r_2 sont des valeurs aléatoires dans $[0;1]$. T représente l'itération en cours et T est le nombre maximal des itérations.

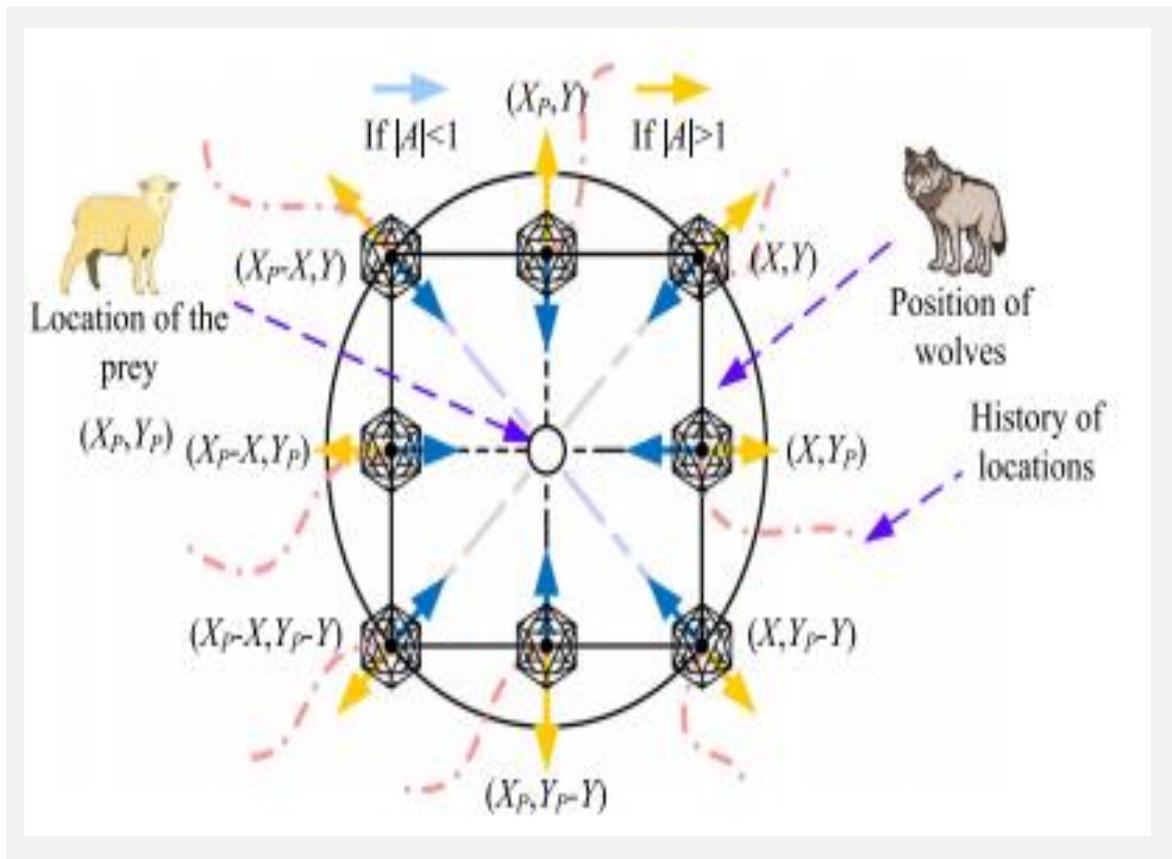


Figure 12 - Encerclement de la proie

3.3.1.2 L'algorithme des faucons (HHO)

Algorithme des faucons (en anglais Harris Hawks Optimisation (HHO)) est un algorithme d'intelligence des essaims basé sur la population qui est proposé par (Heidari et al) [29]. L'inspiration principale de HHO est le comportement coopératif et le style de chasse des faucons de Harris dans la nature, appelés « pounce surprise », également connu sous le nom de stratégie des sept tueries.

Dans cette stratégie intelligente, plusieurs faucons se lancent en coopération sur une proie provenant de différentes directions pour tenter de la surprendre. Les faucons de Harris peuvent révéler une variété de modèles de poursuite basés sur la nature dynamique des scénarios et les modèles d'évasion de la proie. Ce travail imite mathématiquement les modèles et comportements dynamiques pour développer un algorithme d'optimisation.

- **Optimisation des faucons de Harris (HHO) :** L'algorithme HHO est une technique d'optimisation basée sur la population et il peut donc être appliqué à tout problème d'optimisation sous réserve d'une formulation adéquate. **La figure 11** montre toutes les phases du HHO (Exploration et Exploitation).

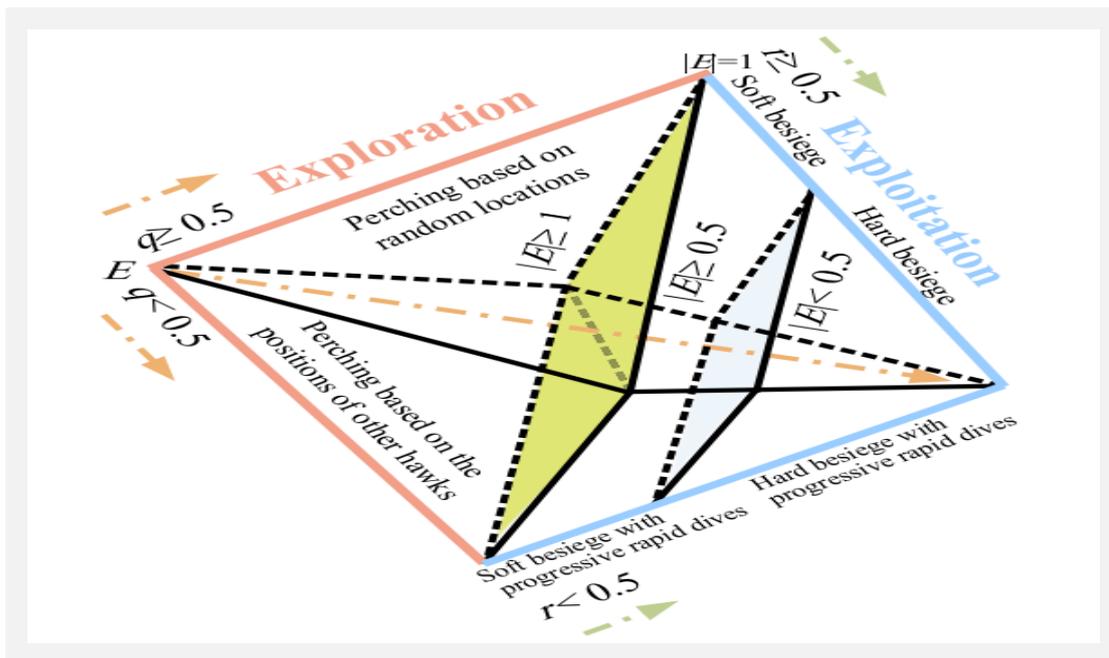


Figure 13- Les différentes phases de HHO

- **Phase d'exploration** : Dans la nature, les faucons de Harris ont des yeux aiguisés qui peuvent les aider à surveiller et à découvrir la proie. Dans cette partie, le mécanisme d'exploration du HHO est proposé, alors les faucons de Harris représentent les solutions, tandis que la proie représente la meilleure solution dans chaque itération. Les faucons de Harris s'installent au hasard dans certains endroits et ils ont deux stratégies pour attaquer leurs proies.

$$X_{\text{rand}}^{t+1} = \begin{cases} X_{\text{rand}}^t - r_1 |2r_2 X^t| & \text{si } q \geq 0.5 \\ (X_{\text{rabbit}}^t - X_m^t) - r_3 (LB + r_4 (UB - LB)) & \text{si } q \leq 0.5 \end{cases}$$

3.5

Avec r_1, r_2, r_3, r_4 et q sont des nombres aléatoires qui appartiennent à $[0,1]$.

- La première règle de l'équation, représente la génération aléatoire de solutions.
- La deuxième règle de l'équation, représente la différence entre la position de la meilleure solution (lapin) et la position moyenne du groupe.

La position moyenne des faucons peut être définie comme indiqué dans **l'équation 3.6** :

$$X_m^t = \frac{1}{N} \sum_{n=1}^N X_i^t$$

3.6

X_i^t : Représente la position de chaque solution (faucon) dans l'itération t.

N : Indique le nombre total de solutions (faucons).

➤ **Phase d'Exploitation** : Les faucons de Harris exécutent une descente rapide en faisant capturer la proie, cependant, les proies ont une puissante capacité à échapper à une situation risquée.

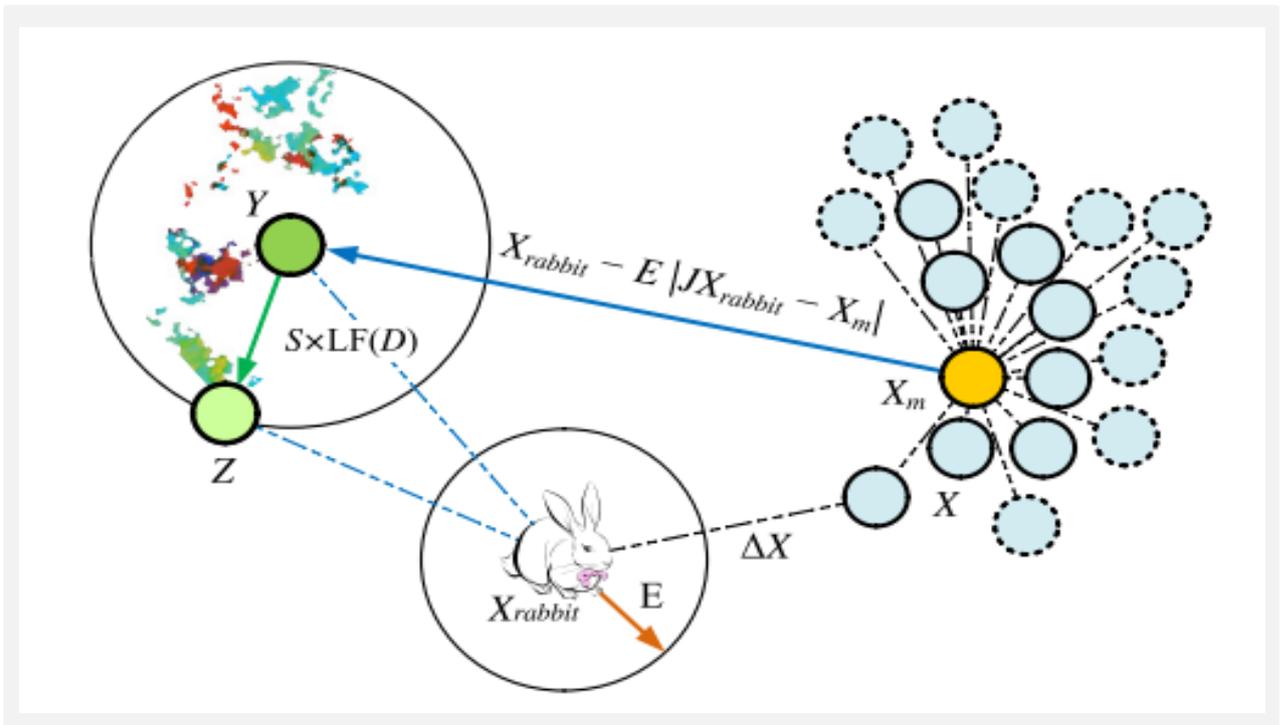


Figure 14-Le processus d'encerclement fort avec une descente progressive rapide dans un espace à 2D.

3.3.2 Les algorithmes inspirés de la physique

3.3.2.1 Algorithme d'optimisation d'Archimède (AOA)

AOA est un algorithme inspiré de la physique et plus particulièrement de la loi d'Archimède. Cet algorithme est introduit par (fatma HASHIM) en 2020 et qui appartient à la classe de méta-heuristiques [30]. La particularité de cet algorithme réside dans le codage de la solution, qui englobe trois informations auditives : volume (**V**), densité (**D**) et accélération(**T**) à base des agents. Donc, initialement, le groupe d'agents est généré aléatoirement dans les dimensions Dim. Comme information supplémentaire, une valeur aléatoire de (**V**), (**D**) et (**T**) sont prévus. Après, le processus d'évaluation est réalisé pour chaque objet afin de déterminer le meilleur objet (O_{Best}) Pendant le processus d'AOA, la mise à jour de la densité et du volume est réalisée afin de changer l'accélération en utilisant le concept de collision entre objets, qui jouent un rôle important pour déterminer la nouvelle position de la solution actuelle. Les étapes générales de l'AOA sont décrites comme suivants :

- **Étape 1 : Processus d'initialisation :** Cette étape vise à initialiser aléatoirement la population réelle qui contient N objets en utilisant **équation (3.7)**. De plus, chaque objet est caractérisé par sa densité (D_i) , son volume (V_i) et l'accélération (T_i) qui sont définies de manière aléatoire à l'aide des équations suivantes

$$o_i = Min_i + r_i * (Max_i - Min_i) ; i = 1, 2, 3, \dots, N$$

3.7

$$D_i = r_i$$

3.8

$$T_i = \text{Min}_i + r_4 * (\text{Max}_i - \text{Min}_i) ; i = 1, 2, 3, \dots, N$$

3.9

Où O_i représente le $i_{\text{ème}}$ objet, Min_i et Max_i sont les bornes maximum et minimum de l'espace de recherche, respectivement. $r_1, r_2, r_3, \text{ et } r_4$ sont des vecteurs aléatoires appartenant à $[0,1]^{Dim}$. La population sera évaluée en calculant l'adéquation de chaque objet afin de déterminer le meilleur objet (O_{Best}) en joignant leurs meilleures valeurs de densité (D_{Best}), de volume (v_{Best}) et accélération (T_{Best}).

- **Étape 2 : La mise à jour des densités et des volumes :** Dans cette étape, les valeurs de densité et le volume de chaque objet sont mis à jour par le contrôle de la meilleure densité et du meilleur volume en utilisant **Équation (3.10)** et **Équation (3.11)**.

$$D_i^{t+1} = D_i^t + S_1 * (D_{Best} - D_i^t)$$

3.10

$$V_i^{t+1} = V_i^t + S_2 * (V_{Best} - V_i^t)$$

3.11

Où S_1, S_2 sont des scalaires aléatoires dans $[0,1]$.

- **Étape 3 : Coefficient de transfert & scalaire de densité :** Dans cette étape, la collision entre les objets se produit jusqu'à l'obtention de l'état d'équilibre. Le rôle principal de la fonction de transfert (T_c) consiste à passer du mode exploration au mode exploitation, défini par **l'équation 3.12** :

$$T_c = \exp\left(\frac{t - T}{T}\right) \quad \text{3.12}$$

(T_c) augmente exponentiellement au fil du temps jusqu'à atteindre 1. t est l'itération courante, tandis que T indique le nombre maximum d'itérations. Aussi, la diminution de la densité scalaire d_s en AOA permet de trouver une solution optimale en utilisant **l'équation (3.13)** :

$$D_s^{t+1} = \exp\left(\frac{t - T}{T}\right) - \left(\frac{t}{T}\right) \quad \text{3.13}$$

- **Étape 4 : Phase d'exploration :** Dans cette étape, la collision entre les agents se produit en utilisant une sélection aléatoire de matériel (M_r), Ainsi, la mise à jour des objets d'accélération est appliquée en utilisant **Équation (3.14)**, lorsque la valeur de la fonction de transfert est inférieure ou égale à 0,5.

$$T_t^{t+1} = \frac{D_{Mr} + V_{Mr} + T_{Mr}}{D_i^{t+1} * V_i^{t+1}} \quad \text{3.14}$$

T_{Best} est l'accélération de l'objet optimal O_{Best} .

- **Étape 6 : Normalisation de l'accélération :** Dans cette étape, nous normalisons l'accélération afin de déterminer le taux de changement à l'aide de l'équation (3.15) :

$$T_{i-norm}^{t+1} = \alpha * \frac{T_i^{t+1} - \min(T)}{\max(T) - \min(T)} + \beta \quad 3.15$$

Où α et β sont fixés respectivement à 0,9 et 0,1. Le T_{i-norm}^{t+1} indique l'accélération normalisée. Une valeur d'accélération inférieure indique que l'objet fonctionne dans le mode d'exploitation ; sinon, l'objet fonctionne en mode exploration.

- **Étape 7 : Le processus de mise à jour :** Pour la phase d'exploration ($T_c \leq 0.5$), la position de $i^{ème}$ objet dans l'itération $t + 1$ est modifiée par l'équation (3.15), tandis que la position de l'objet est mise à jour par Équation (3.16) en phase d'exploitation ($T_c > 0.5$) :

$$O_i^{t+1} = O_i^t + c_1 * r_5 * T_{i-norm}^{t+1} * d_s * (O_{rand} - O_i^t) \quad 3.16$$

3.3.2.2 Algorithme des prédateurs marins (MPA)

L'algorithme des prédateurs marins (MPA) est un algorithme d'optimisation inspiré de la nature qui suit les règles qui régissent naturellement la stratégie de recherche de nourriture optimale et la politique de taux de rencontre entre prédateur et proie dans les écosystèmes marins.

La stratégie de Lévy est un modèle répandu parmi les prédateurs marins (par exemple, les requins, les thons, les marlins) lorsqu'ils recherchent de la nourriture dans un environnement pauvre en proies, mais lorsqu'il s'agit de se nourrir dans une zone riche en proies, le modèle est généralement basculé vers le mouvement brownien [31]. La politique de taux de rencontre optimal dans l'interaction biologique entre prédateur et proie dépend également du type de mouvement que chacun des prédateurs/proies effectue et du rapport de vitesse de la proie au prédateur [32].

Les faits saillants suivants résument les politiques régissant la recherche de nourriture, les interactions et les souvenirs optimaux chez les prédateurs marins :

- Les prédateurs marins utilisent la stratégie de Lévy pour l'environnement à faible concentration de proies tout en employant le mouvement brownien pour les zones à proies abondantes.
- Ils ont montré les mêmes pourcentages de mouvement Lévy et Brownien au cours de leur vie de traversée de différents habitats.

- En raison des effets environnementaux tels que les effets naturels (formation de tourbillons) ou d'origine humaine (DCP), ils modifient leur comportement pour, espérons-le, trouver des zones avec une répartition différente des proies.
- En faible rapport de vitesse $v = (0,1)$, la meilleure stratégie pour un prédateur est Lévy ; soit la proie se déplace en brownien ou en Lévy.
- Dans le rapport de vitesse unitaire ($v = 1$), si la proie se déplace à Lévy, la meilleure stratégie pour un prédateur est brownienne. D'autres scénarios dépendent de la taille du système.
- Dans un rapport de vitesse élevé ($v \geq 10$), la meilleure stratégie pour un prédateur est de ne pas bouger du tout. Dans ce cas, soit la proie se déplace Brownian ou Lévy.
- Ils profitent d'une bonne mémoire pour rappeler leurs associés ainsi que l'emplacement de la recherche de nourriture réussie.

Sur la base des faits saillants, le processus d'optimisation de l'AMP est divisé en trois phases principales d'optimisation prenant en compte différents rapports de vitesse et imitant en même temps la vie entière d'un prédateur et d'une proie : (1) dans un rapport de vitesse élevé ou lorsque la proie se déplace plus rapidement que le prédateur , (2) dans le rapport de vitesse unitaire ou lorsque le prédateur et la proie se déplacent presque au même rythme, et (3) dans le rapport de vitesse faible lorsque le prédateur se déplace plus rapidement que la proie.

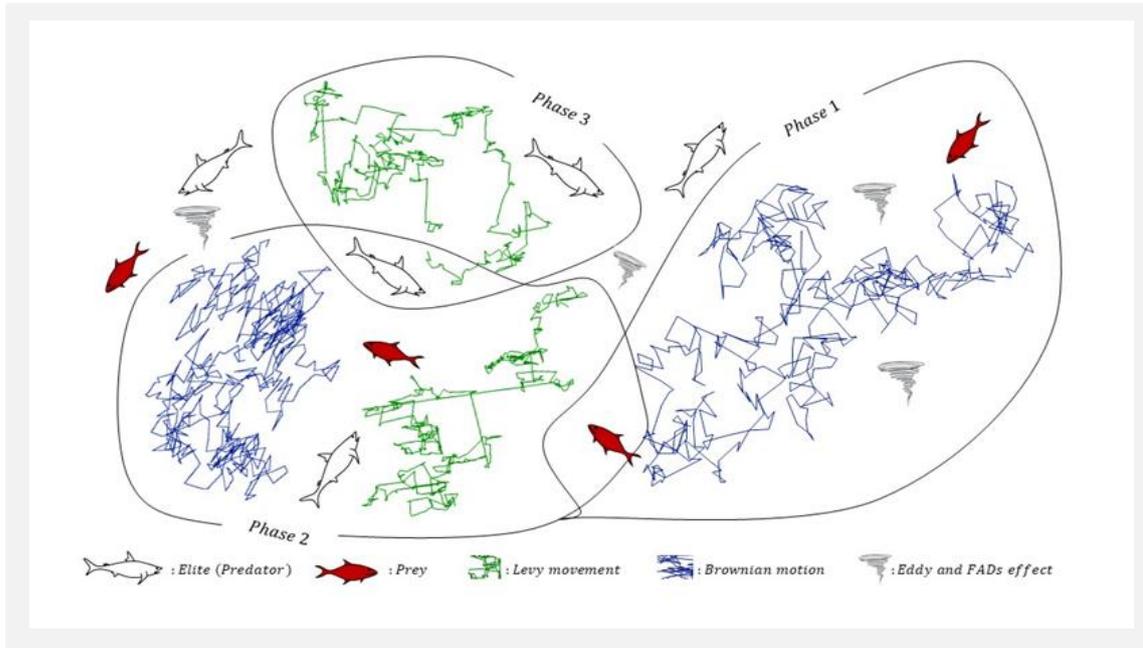


Figure 15 - La stratégie de Lévy (MPA)

3.3.3 Structure de l'objet immergé

Cet ingrédient joue un rôle essentiel dans le processus d'optimisation à base des algorithmes inspirés de la physique ou de la théorie d'intelligence distribuée. La taille de chaque objet correspond au nombre de blocs, qui est un vecteur unidimensionnel « 1D » composé de 49 éléments. Les éléments du vecteur sont générés aléatoirement dans la plage [0, 1]. À ce stade, si la valeur est supérieure ou égale à "0.5", la valeur est donc arrondie à "1". Dans ce cas, le bloc est considéré comme une caractéristique pertinente, qui est codé par un histogramme à base de LBP, HOG ou un vecteur de paramètres à base de GLCM. Au contraire, le bloc sera ignoré lorsque la valeur est autour de zéro. Ce vecteur sera transformé à une matrice de taille 7*7 afin de projeter la solution d'encodage sur l'image pour extraire les histogrammes à base

des blocs multiples. Cet encodage vise à sélectionner les zones informatives en réalisant la concaténation des histogrammes à base de plusieurs blocs (LBP, LBP-52, HOG ou GLCM).

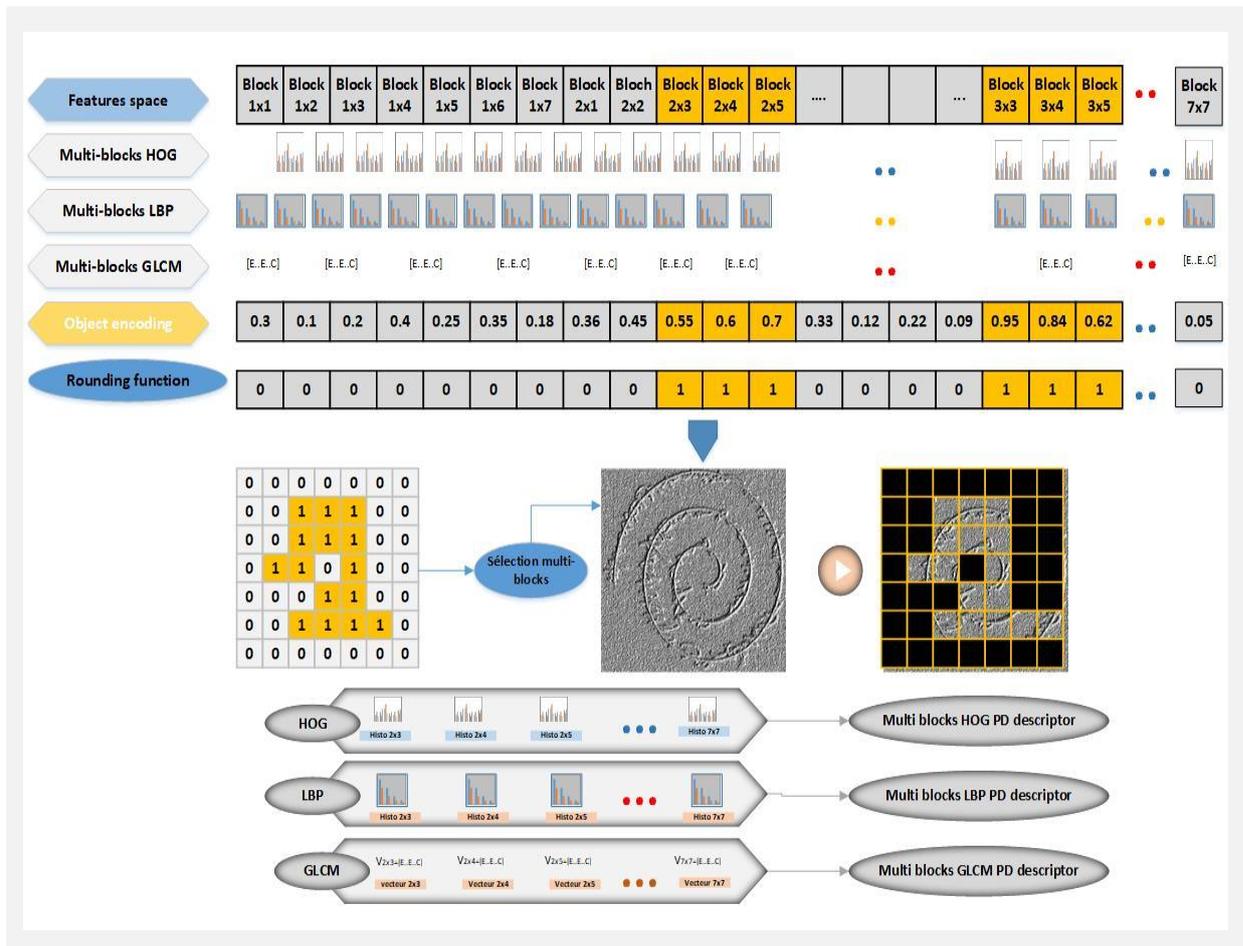


Figure 16 – Prétraitement des images

3.4 Conclusion

Dans ce chapitre, on a présenté dans un premier temps de manière générale les différentes inspirations utilisées. Par la suite, on a présenté d'une manière plus avancée les algorithmes évolutionnaires en détaillant leurs différentes composantes ainsi que le déroulement du processus d'optimisation. Les algorithmes d'optimisation incluant plusieurs inspirations comme l'inspiration à base de l'intelligence distribuée, l'inspiration mathématique et une simulation des lois de la physique. Cette richesse a engendré la création de plusieurs algorithmes qui ont été expliqués en détail, incluant l'algorithme des loups gris (GWO), l'algorithme des faucons (HHO), Algorithme des prédateurs marins (MPA).

Dans le prochain chapitre on va citer les mesures de performances, discuter les résultats obtenus.

Chapitre 4

Résultats Expérimentaux & Discussion

4.1 Introduction

Cette partie du manuscrit inclut nos réalisations à la détection de la maladie de parkinson et détecter le trouble à partir des images spirales. La comparaison entre l'approche proposée avec d'autres algorithmes bio-inspirés, nécessite de satisfaire un certain nombre de conditions.

Premièrement, certains critères doivent être mesurés pour toutes les méthodes. Deuxièmement, on va spécifier certains ingrédients comme le choix du classifieur, Ensuite, le choix d'un algorithme bio-inspiré pose un vrai challenge car il y'a plusieurs inspirations comme l'inspiration génétique, l'inspiration de l'intelligence distribuée et l'inspiration à base des lois physiques et mathématiques. Dans notre étude, on va choisir de tester les algorithmes englobant HHO, AOA et GWO, MPA. Concernant les techniques d'extraction, on a distingué trois méthodes dans la littérature LBP, HOG et GLCM, avec le classifieur KNN, Après on va choisir le meilleur résultat obtenus (best) pour les descripteurs et pour les algorithmes et on va appliquer le classifieur MLP ou bien SVM.

Enfin, une analyse globale a été introduite pour tirer des conclusions concernant les approches proposées.

4.2 Les mesures de performance

Les mesures de performances qui sont employées afin d'évaluer la qualité de la détection de la maladie de parkinson à travers les multiples combinaisons d'entraînement. Les différentes manières d'application de ces mesures y sont également discutées.

Différentes mesures d'évaluation ont été proposées afin d'évaluer les performances des différents algorithmes d'apprentissage. L'évaluation empirique des algorithmes est une question qui occupe plusieurs chercheurs. La plupart des mesures utiliser dans notre travail se concentrent sur la capacité des classifieurs à identifier correctement les observations de chaque classe.

Les mesures de qualité utilisées dans notre travail sont : Accuracy (Acc), Rappel (Re), Précision (Pr) et F-score (F- score).

On va utiliser toutes ces mesures pour comparer les approches HHO, GWO, AOA, MPA pour le problème de détection de la maladie de parkinson via des algorithmes bio-inspirés.

- **Matrice de confusion :** La qualité d'un système de classification est mesurée à l'aide de la matrice de confusion. Les lignes de cette matrice représentent la répartition des objets dans les classes réelles. Les colonnes quant à elles, représentent la répartition des points dans les classes estimées par un algorithme de classification. Le tableau montre les paramètres quantitatifs dérivés de la matrice de confusion.

Classes estimées		
Réelles	Oui	Non
Oui	TP	FN
Non	FP	TN

Tableau 1 - Matrice de confusion

Les mesures de performance les plus connues dans le contexte de la recherche d'information (précision, rappel, la F-score, TP, FP) seront utilisées pour évaluer la performance de notre approche.

- TP (True Positive) : Le classifieur arrive à identifier la personne comme étant une personne atteinte de la maladie.
- TN (True Négative) : le classifieur arrive à identifier la personne comme étant une personne en bonne santé.
- FP (False Positive) : le classifieur identifie la personne comme étant une personne atteinte de la maladie sachant que l'étiquetage réel indique que la personne est en bonne santé.
- FN (True Négative) : le classifieur identifie la personne comme étant une personne en bonne santé sachant que l'étiquetage réel indique que la personne atteinte la maladie.

- **Taux de reconnaissance (Accuracy) :** Cette mesure signifie le pourcentage des personnes malade et des personnes en bonne santé qui sont correctement classifiées, définie par l'équation suivante :

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TN} \quad 4.1$$

- **Sensibilité (Taux de rappel, Recall en anglais) :** Cette mesure est aussi appelée taux de vrais positifs (TPR), qui indique le pourcentage de la prédiction des hommes en utilisant la formule suivante :

$$Rappel = \frac{TP}{TP + FN} \quad 4.2$$

- **Précision :** La précision est la proportion parkinsonisme au niveau de la prédiction obtenue par le classifieur :

$$Précision = \frac{TP}{TP + FP} \quad 4.3$$

- **La métrique de la valeur de fitness (Sc) :** Cette métrique évalue les performances des algorithmes, qui vise à maximiser le taux de reconnaissance et le taux d'élimination des caractéristiques non pertinentes calculées par :

$$Sc = 0.99 * Acc + 0.01 * \frac{(D - d)}{D} \quad 4.4$$

D est le nombre total d'histogrammes multi-blocs extraits de l'image originale.

- **La taille des entités sélectionnées (Sr) :** Cette métrique représente les blocs sélectionnés dans une image. Il est calculé comme :

$$S = \frac{d}{D} \quad 4.5$$

Où d est le nombre de zones sélectionnées qui augmentent la performance.

- **F-score (F_{sc}) :** En statistique, le F-score indique la moyenne harmonique entre le rappel et la précision. Cette mesure est illustrée dans l'équation (4.6) :

$$F_{sc} = 2 * \frac{R * P}{R + P} \quad 4.6$$

➤ **Temps CPU (Cpu) :** C'est le temps d'extraction requis pour chaque algorithme.

4.3 Analyse graphique

Le graphe de la **figure 17** montre les résultats des mesures de performance de chaque descripteur, on remarque :

- Pour la valeur de précision, le descripteur LBP-52 présente la plus haute avec 91.41%
- Pour la valeur Recall, présente une même valeur pour GLCM et HOG avec 84.93% suite par LBP et LBP-52 avec des valeurs de 90.41%
- Pour F-score, LBP présente une valeur avec 89.46% et LBP-52 avec 88.90% suite par HOG et GLCM avec des valeurs de 81.46% et 80.03%
- Donc, d'après le prétraitement des images avec les descripteurs, LBP-52 prend les plus grandes valeurs de mesures de performances.

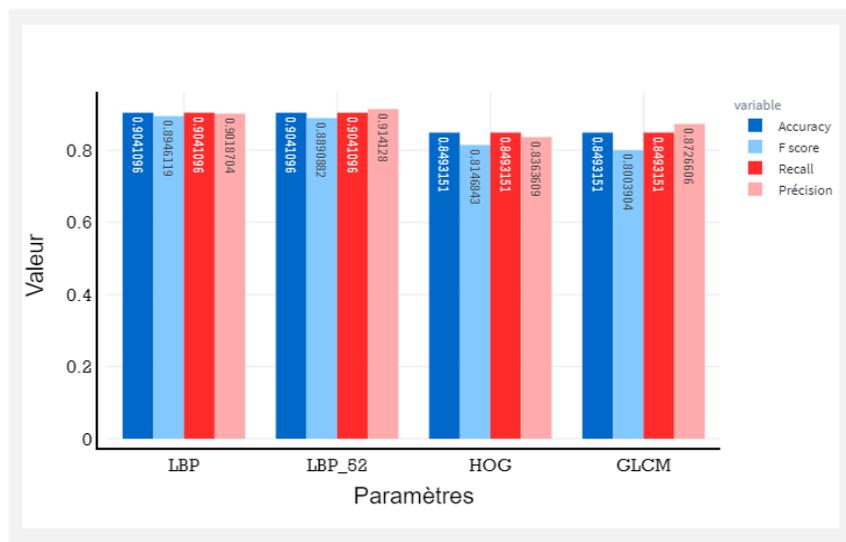


Figure 17- les résultats des mesures de performances de chaque descripteurs

4.3.1 Résultats des descripteurs pour chaque algorithme

4.3.1.1 En terme de fitness et taux de reconnaissance

Tableau 2 résume les résultats des algorithmes à base de la sélection des zones en terme de fitness, en variant les méthodes d'extraction (HOG, LBP, LBP-52 & GLCM) avec le classifieur (KNN) tester sur la base de données (HandPD), on remarque une meilleure performance obtenue par l'Algorithme AOA et MPA sur le descripteur LBP-52 qui atteint une valeur de fitness la plus élevée avec 97.14%.

Fitness / (KNN)	(HandPD)			
	Algorithmes	LBP	LBP_52	HOG
GWO	0,94395342	0,95751507	0,90446849	0,91783014
HHO	0,94115342	0,94115342	0,90286849	0,89110685
AOA	0,95771507	0,97147671	0,93059178	0,91763014
MPA	0,95691507	0,97147671	0,91663014	0,90446849

Tableau 2 -Résultats des descripteurs pour chaque algorithme en terme de fitness

Les graphes suivants montre les résultats des algorithmes en terme de fitness pour chaque descripteur :

- **Figure 18** montre les résultats de descripteur LBP en terme de fitness a base des algorithmes (AOA, MPA, GWO, HHO), on remarque une meilleure performance obtenue par l'Algorithme AOA atteint une valeur de fitness 95.77%.

- **Figure 19** montre les résultats de descripteur LBP-52 en terme de fitness a base des algorithmes (AOA, MPA, GWO, HHO), on remarque une meilleure performance obtenue par l'Algorithme AOA et MPA atteint une valeur de fitness 97.17%.
- **Figure 20** montre les résultats de descripteur HOG en terme de fitness a base des algorithmes (AOA, MPA, GWO, HHO), on remarque une meilleure performance obtenue par l'Algorithme AOA atteint une valeur de fitness 93.05%.
- **Figure 21** montre les résultats de descripteur GLCM en terme de fitness a base des algorithmes (AOA, MPA, GWO, HHO), on remarque une meilleure performance obtenue par l'Algorithme GWO atteint une valeur de fitness 91.78%.

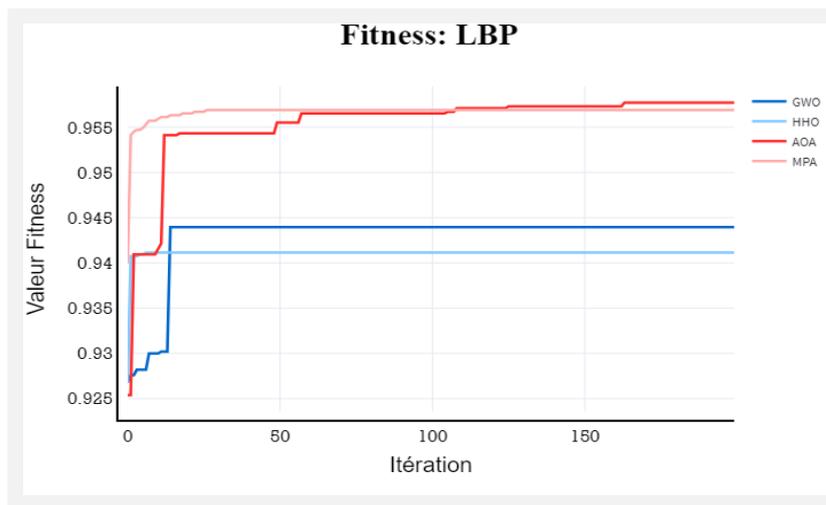


Figure 18– Résultats de LBP en terme de fitness

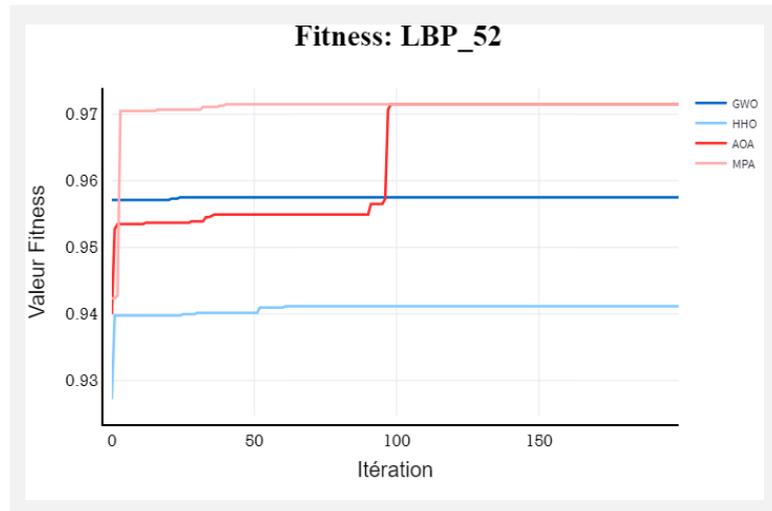


Figure 19- Résultats de LBP-52 en terme de fitness

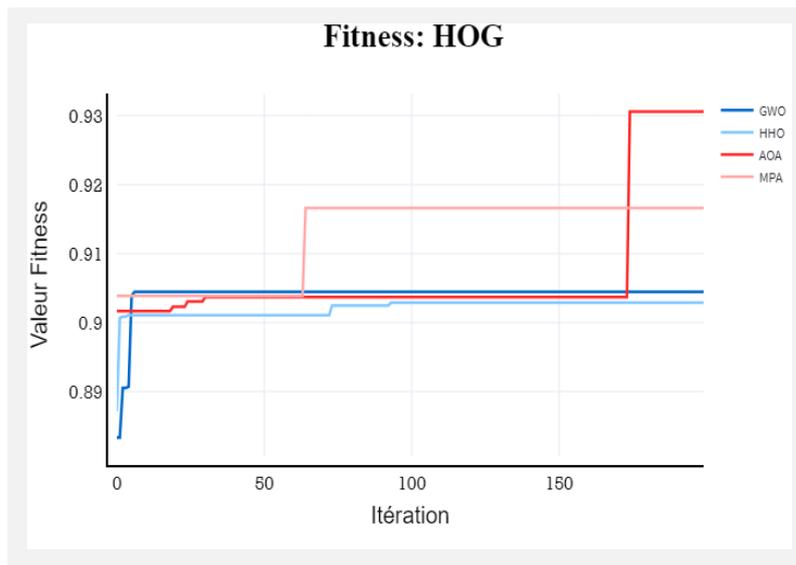


Figure 20- Résultats de HOG en terme de fitness

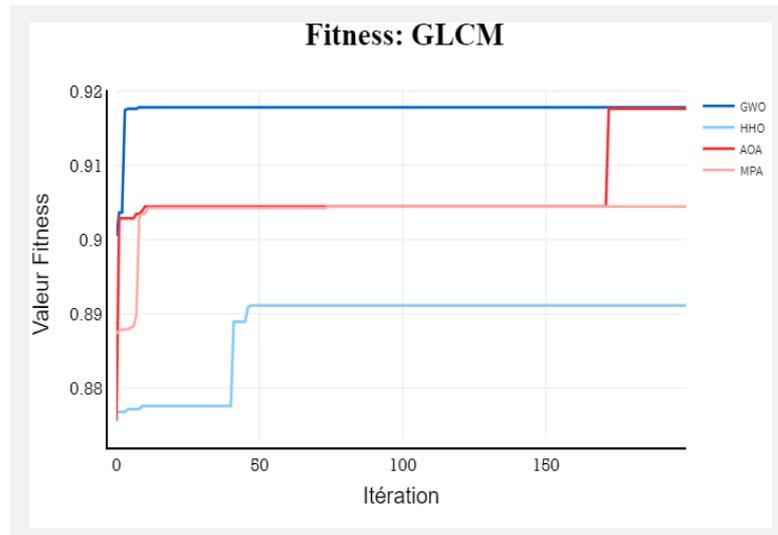


Figure 21- Résultats de GLCM en terme de fitness

Tableau 3 résume les résultats des algorithmes à base de la sélection des zones en termes de taux de reconnaissance, en variant les méthodes d'extraction (HOG, LBP, LBP-52 & GLCM) avec le classifieur (KNN) tester sur la base de données (HandPD), on remarque que la meilleure performance obtenue par l'Algorithme AOA et MPA sur le descripteur LBP-52 qui atteint une valeur de reconnaissance la plus élevée avec 97.26%.

Accuracy/(KNN)	(HandPD)			
	LBP	LBP_52	HOG	GLCM
GWO	0,94395342	0,95890411	0,90410959	0,91780822
HHO	0,94520548	0,94520548	0,90410959	0,89041096
AOA	0,95890411	0,97260274	0,93150685	0,91780822
MPA	0,95890411	0,97260274	0,91780822	0,90410959

Tableau 3 - Résultats des descripteurs pour chaque algorithme en termes d'accuracy.

Les graphes suivants montre les résultats des algorithmes en terme de taux de reconnaissance pour chaque descripteur :

- **Figure 22** montre les résultats de descripteur LBP en terme de de taux de reconnaissance a base des algorithmes (AOA, MPA, GWO, HHO), on remarque une meilleure performance obtenue par l'Algorithme AOA et MPA atteint une valeur de taux de reconnaissance 95.89%.
- **Figure 23** montre les résultats de descripteur LBP-52 en terme de de taux de reconnaissance a base des algorithmes (AOA, MPA, GWO, HHO), on remarque une meilleure performance obtenue par l'Algorithme AOA et MPA atteint une valeur de taux de reconnaissance 97.26%.
- **Figure 24** montre les résultats de descripteur HOG en terme de de taux de reconnaissance a base des algorithmes (AOA, MPA, GWO, HHO), on remarque une meilleure performance obtenue par l'Algorithme AOA atteint une valeur de taux de reconnaissance 93.15%.
- **Figure 25** montre les résultats de descripteur GLCM en terme de de taux de reconnaissance a base des algorithmes (AOA, MPA, GWO, HHO), on remarque une meilleure performance obtenue par l'Algorithme AOA et GWO atteint une valeur de taux de reconnaissance 91.78%.

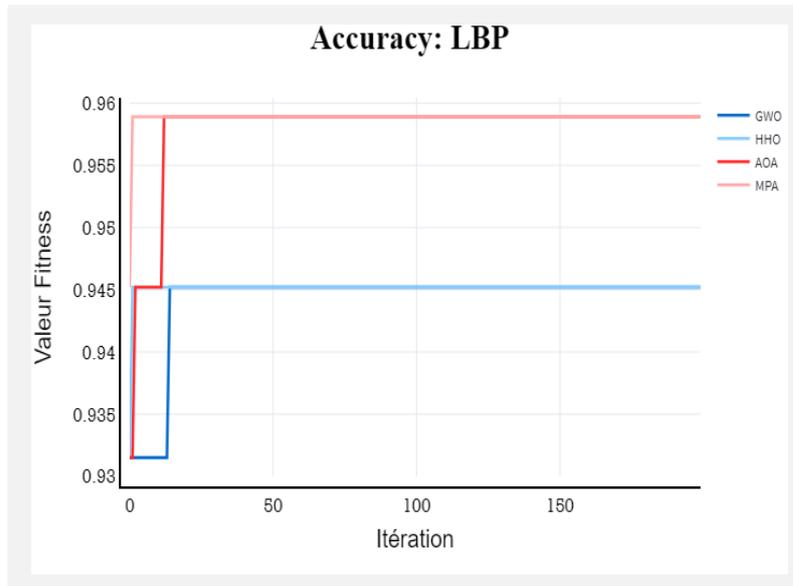


Figure 22 - Résultats de LBP en terme d'accuracy

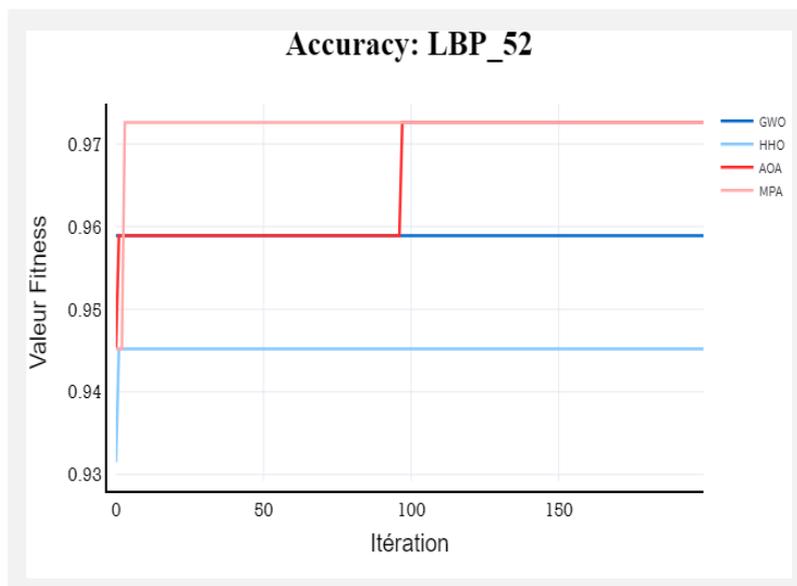


Figure 23- Résultats de LBP-52 en terme d'accuracy

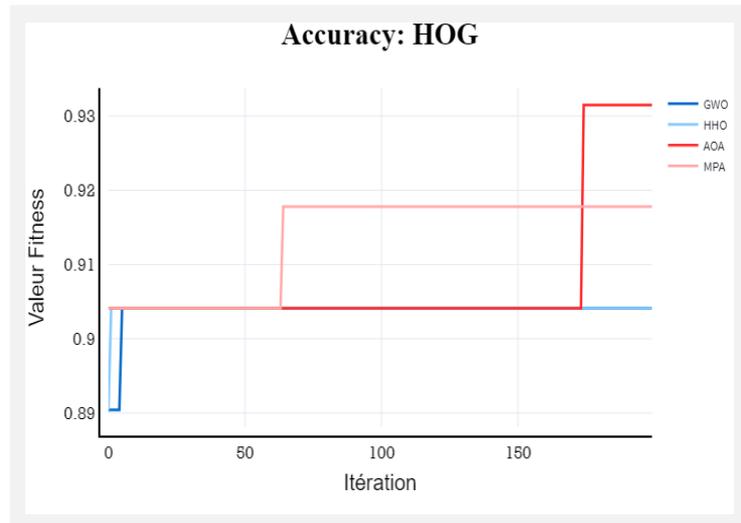


Figure 24 - Résultats de HOG en terme d'accuracy



Figure 25 - Résultats de GLCM en terme d'accuracy

Les graphes suivants montre la comparaison des descripteurs a base des algorithmes en terme de fitness et de taux de reconnaissance, on remarque :

- Les algorithmes MPA, AOA sur le descripteur LBP-52 présentent les taux de fitness les plus élevés avec 97.14%.
- Les algorithmes MPA, AOA sur le descripteur LBP-52 présentent les taux de reconnaissance les plus élevés avec 97.26%.

Les algorithmes HHO, GWO sur tous les descripteurs présentent les taux de reconnaissance et de fitness les moins élevés.

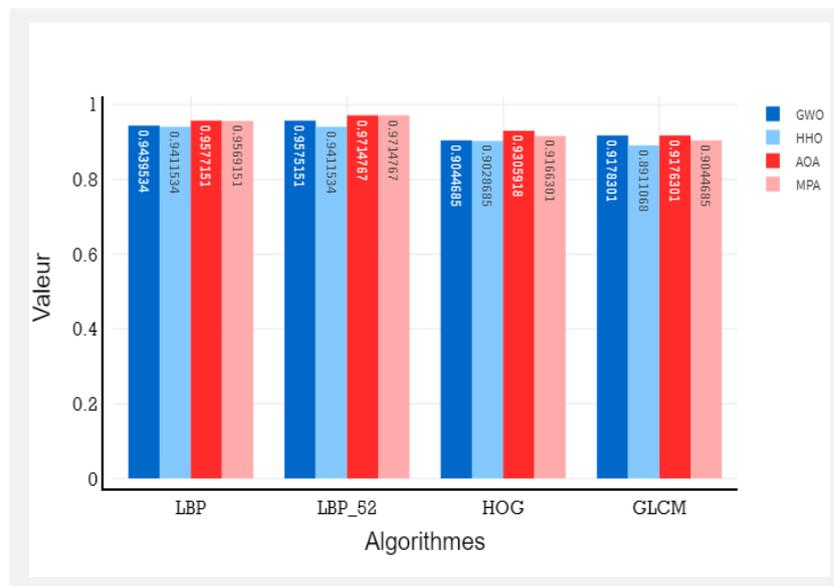


Figure 26 - la comparaison des algorithmes en terme de fitness

4.3.2 Résultats des algorithmes pour chaque descripteur

4.3.2.1 En terme de fitness et taux de reconnaissance

Les graphes suivants montre les résultats des algorithmes en terme de fitness pour chaque descripteur :

- **Figure 27** montre les résultats d'algorithme GWO en terme de fitness a base des descripteurs (LBP, LBP-52, GLCM, HOG), on remarque une meilleure performance obtenue par le descripteur LBP-52 atteint une valeur de fitness 95.75%.
- **Figure 28** montre les résultats d'algorithme HHO en terme de fitness a base des descripteurs (LBP, LBP-52, GLCM, HOG), on remarque une meilleure performance obtenue par les descripteurs LBP-52 et LBP atteint une valeur de fitness 94.12%.
- **Figure 29** montre les résultats d'algorithme AOA en terme de fitness a base des descripteurs (LBP, LBP-52, GLCM, HOG), on remarque une meilleure performance obtenue par le descripteur LBP-52 atteint une valeur de fitness 97.15%.
- **Figure 30** montre les résultats d'algorithme MPA en terme de fitness a base des descripteurs (LBP, LBP-52, GLCM, HOG), on remarque une meilleure performance obtenue par le descripteur LBP-52 atteint une valeur de fitness 97.15%.

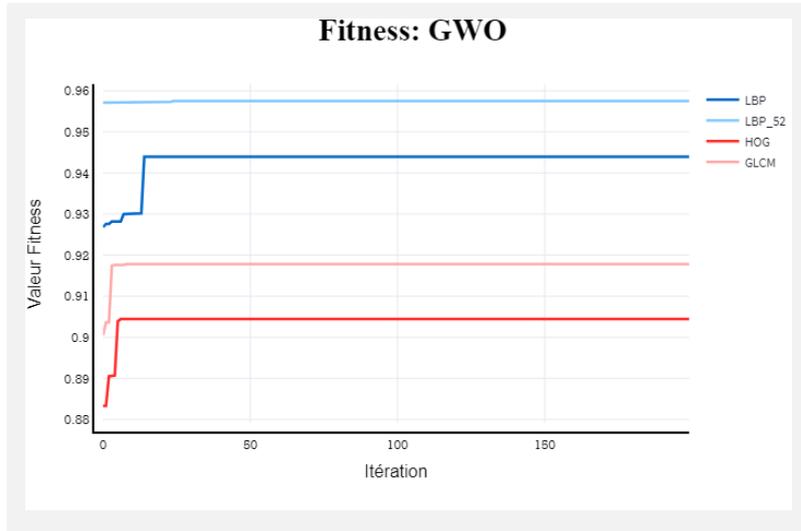


Figure 27 - Résultats de GWO en terme de fitness

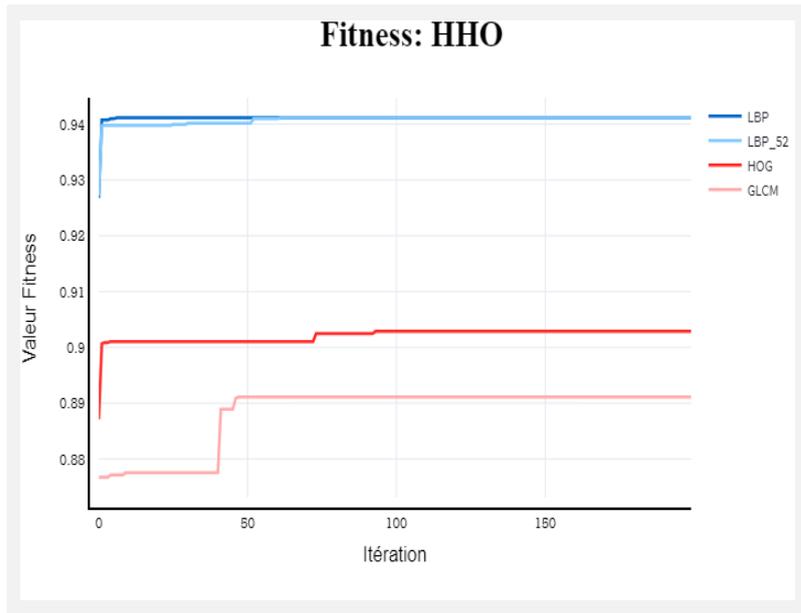


Figure 28 - Résultats de HHO en terme de fitness

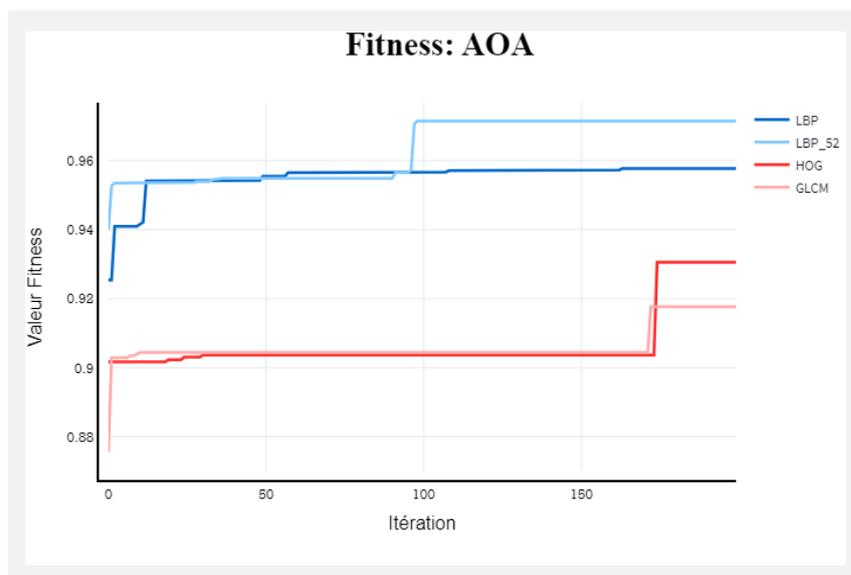


Figure 29 – Résultats de AOA en terme de fitness

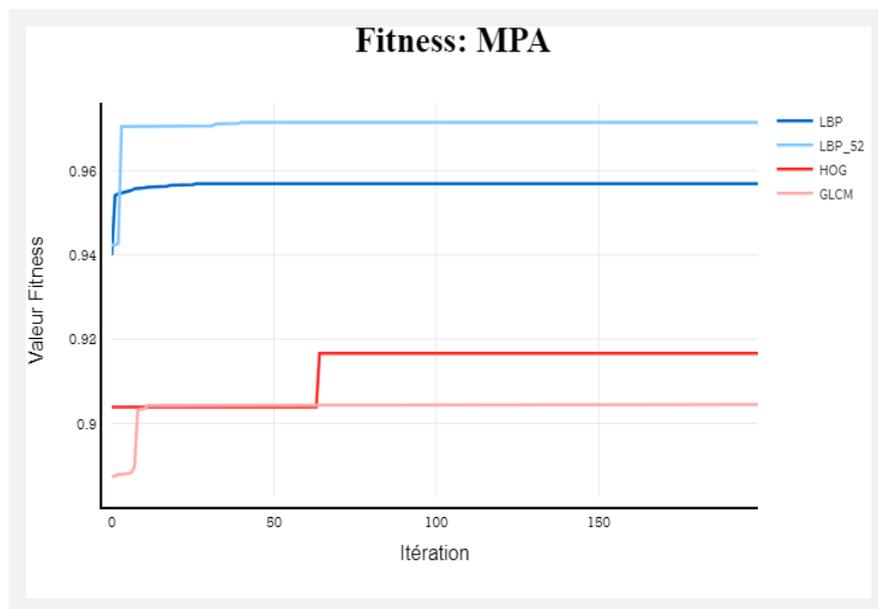


Figure 30 - Résultats de MPA en terme de fitness

Les graphes suivants montre les résultats des descripteurs en terme de taux de reconnaissance pour chaque algorithme :

- **Figure 31** montre les résultats d'algorithme GWO en terme de taux de reconnaissance a base des descripteurs (LBP, LBP-52, GLCM, HOG), on remarque une meilleure performance obtenue par le descripteur LBP-52 atteint une valeur de taux de reconnaissance 95.89%.
- **Figure 32** montre les résultats d'algorithme HHO en terme de taux de reconnaissance a base des descripteurs (LBP, LBP-52, GLCM, HOG), on remarque une meilleure performance obtenue par les descripteurs LBP-52 et LBP atteint une valeur de taux de reconnaissance 94.52%.
- **Figure 33** montre les résultats d'algorithme AOA en terme de taux de reconnaissance a base des descripteurs (LBP, LBP-52, GLCM, HOG), on remarque une meilleure performance obtenue par le descripteur LBP-52 atteint une valeur de taux de reconnaissance 97.26%.
- **Figure 34** montre les résultats d'algorithme MPA en terme de fitness a base des descripteurs (LBP, LBP-52, GLCM, HOG), on remarque une meilleure performance obtenue par le descripteur LBP-52 atteint une valeur de taux de reconnaissance 97.26%.

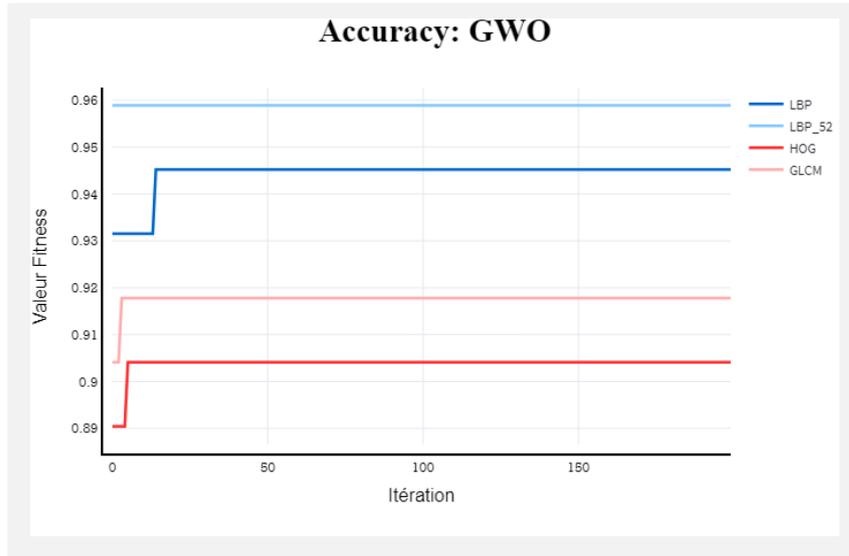


Figure 31 - Résultats de GWO en terme d'accuracy

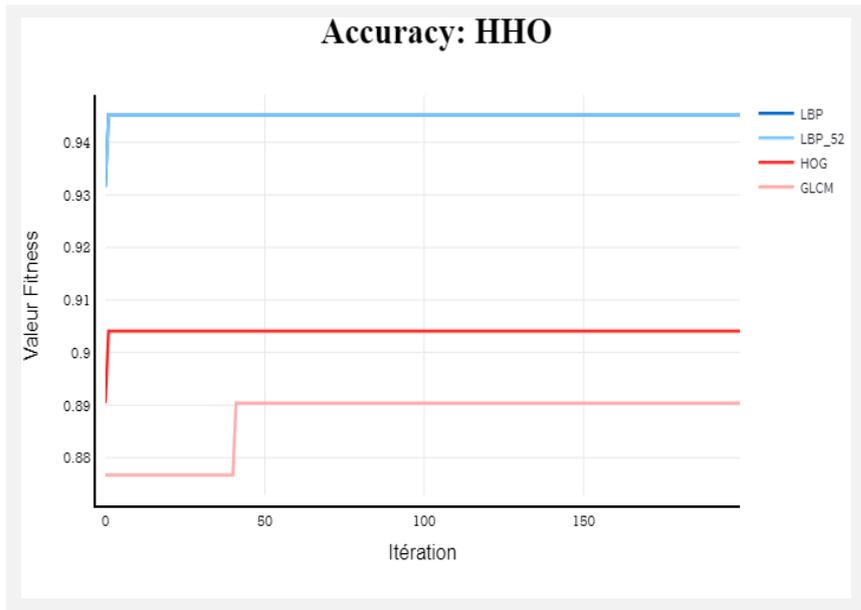


Figure 32 - Résultats de HHO en terme d'accuracy

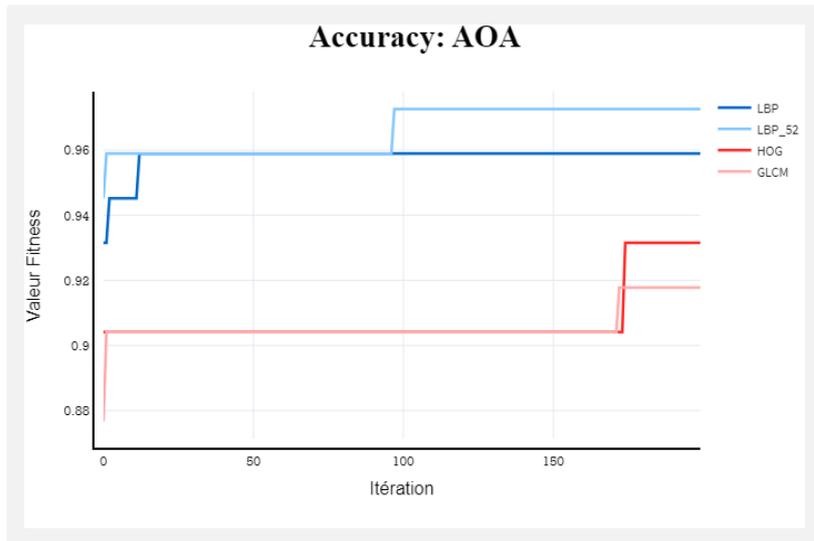


Figure 33- Résultats de AOA en terme d'accuracy

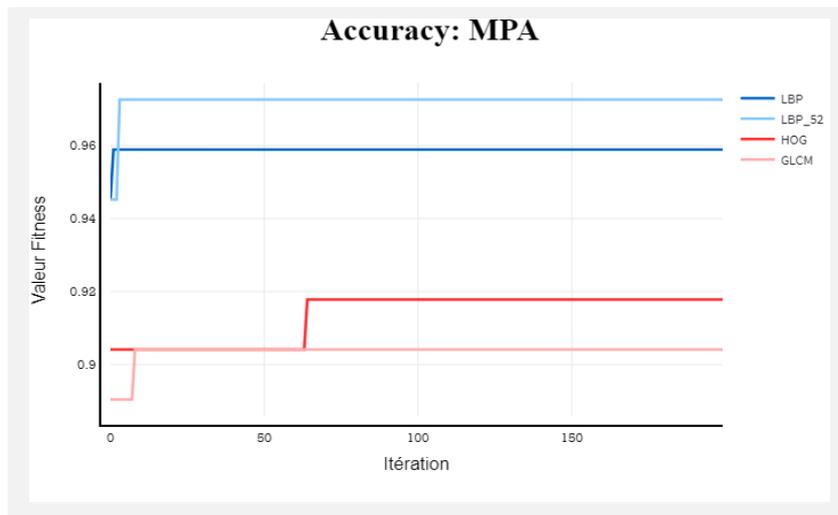


Figure 34- Résultats de MPA en terme d'accuracy

Alors, on conclut que le descripteur le plus performant est "LBP-52" avec l'algorithme AOA a base de classifieur KNN, après cette optimisation on va prendre les résultats obtenus pour les tester avec les deux autre classifieur (MLP, SVM).

4.3.3 Résultats pour les deux classifieurs (MLP, SVM)

Le graphe de la **figure 35** montre les mesures de performances de deux classifieurs, on remarque :

- Pour la valeur de précision, le classifieur MLP présente la plus haute avec 81.43%.
- Pour la valeur de Recall, le classifieur MLP présente la plus haute avec 98.37%.
- Pour la valeur de F-score, le classifieur MLP présente la plus haute avec 88.87%.

Donc, d'après la fonction de fitness, on remarque une meilleure performance obtenue par le classifieur MLP atteint une valeur de taux de reconnaissance 98.37%.

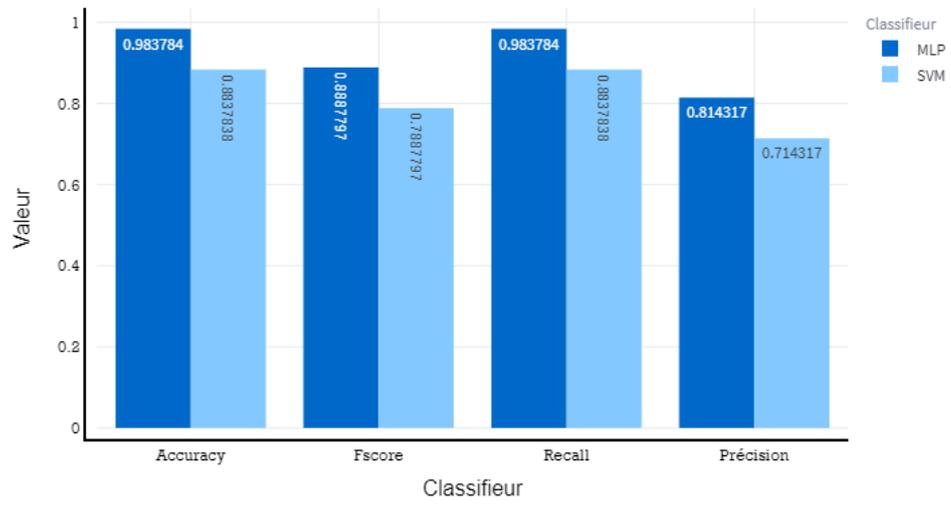


Figure 35 - les résultats des mesures de performances de chaque classifieurs

4.4 Conclusion

Dans ce chapitre, on a montré le coté expérimental d'un nouveau concept de la détection de la maladie à base des algorithmes d'optimisation.

Les étapes réalisées, sont les suivantes :

- Premièrement, après la réalisation des prétraitements sur les images avec les quatre descripteurs (LBP, LBP-52, HOG, GLCM) on a obtenu que LBP-52 est le descripteur le plus performant avec une précision 90.41% avec un classifieur KNN.
- Deuxièmement, Les résultats expérimentaux ont montré l'efficacité de l'algorithme AOA et MPA a base de LBP-52 avec une précision 97.14% par rapport à d'autres concurrents.
- Troisièmement, parmi les deux algorithmes AOA, MPA, on a choisi AOA qui a obtenu la valeur la plus élevé de taux de reconnaissance avec moins de nombre de paramètre, tous les résultats obtenus ont été basées sur la fonction fitness.
- Quatrièmement, après l'optimisation on a pris les meilleurs résultats du classifieur KNN qui sont : le descripteur LBP-52 et l'algorithme d'optimisation AOA, on a appliqué sur ces derniers les deux classifieurs MLP, SVM, le classifieur MLP est le plus performant parmi les trois classificateurs qui atteint la plus grande précision avec 98.37%.

Conclusion Générale

Au niveau de cette étude, on a exploré plusieurs domaines d'applications à la fois en commençant par la détection automatique de la maladie, la sélection des zones et l'utilisation des techniques d'optimisation avancées.

Il est important de souligner que la détection automatique de la maladie de parkinson via des algorithmes bio-inspirés nécessite la phase d'extraction des zones à partir des images de tremblement des mains contenant des dessins (spirale) de plusieurs individus avec la base des données (HandPD).

Le prétraitement a été réalisé en utilisant quatre descripteurs : HOG, LBP et GLCM, LBP-52 et concernant le problème de la dimensionnalité a été résolu en utilisant plusieurs approches inspirées de la théorie de Swarm Intelligence, de la physique et de mathématique.

En plus, les algorithmes AOA et MPA arrivent à sélectionner un nombre réduit de zones pour réaliser la détection qui est devenue un sujet d'actualité.

Pour l'ensemble de donnée (HandPD) qui contient plus de 29000 images, l'algorithme AOA à base des quatre descripteurs (HOG, LBP et GLCM, LBP-52) ont montré leurs efficacités en comparant avec d'autres algorithmes comme GWO, MPA et HHO.

D'après les résultats expérimentaux, on a constaté que l'algorithme AOA-LBP-52 a obtenu une meilleure performance en termes de taux de reconnaissance 97.14% à base de classifieur KNN.

L'algorithme AOA-LBP-52 a prouvé sa robustesse au niveau de la détection de la maladie avec un taux de 98.37% en utilisant la validation croisée à l'aide d'un perceptron multicouche. AOA-LBP-52 détermine seulement 7 zones pour distinguer la maladie de parkinson.

Durant ce travail on a trouvé que c'est un sujet intéressant qui nécessite plus de recherche et d'amélioration.

Alors, pour ce qu'ils veulent traiter le même sujet on les conseille de suivre un autre plan de travail ce qui est :

- Il est préférable d'ajouter plus des zones pour plus de précision.
- Utiliser d'autre descripteur comme les filtres de haar, (HOI).
- Utiliser d'autre algorithmes d'optimisations comme (EO, HGSO, EPO, MRFO, SCA).
- Combiner entre deux descripteurs et plus.

Enfin, on souhaite vraiment que ce sujet va prendre l'importance qu'il mérite.

Bibliographie

- [1] L. M. De Lau and M. M. Breteler. Epidemiology of Parkinson's disease. *The Lancet Neurology*, 5(6) :525–535, 2006.
- [2] A. Samii, J. G. Nutt, and B. R. Ransom. Parkinson's disease. *The Lancet*, 363(9423) :1783 – 1793, 2004.
- [3] A. Ascherio, H. Chen, M. G. Weisskopf, E. O'Reilly, M. L. McCullough, E. E. Calle, M. A. Schwarzschild, and M. J. Thun. Pesticide exposure and risk for parkinson's disease. *Annals of Neurology : Official Journal of the American Neurological Association and the Child Neurology Society*, 60(2) :197–203, 2006.
- [4] X. Mao, M. T. Ou, S. S. Karuppagounder, T.-I. Kam, X. Yin, Y. Xiong, P. Ge, G. E. Umanah, S. Brahmachari, J.-H. Shin, et al. Pathological α -synuclein transmission initiated by binding lymphocyte-activation gene 3. *Science*, 353(6307) :aah3374, **2016**.
- [5] Johann Faouzi. Machine learning to predict impulse control disorders in Parkinson's disease. *Artificial Intelligence [cs.AI]*. Sorbonne Université, **2020**. English. NNT : 2020SORUS048.
- [6] Sakar, B.E.; Isenkul, M.E.; Sakar, C.O.; Sertbas, A.; Gurgun, F.; Delil, S.; Apaydin, H.; Kursun, O. Collection and analysis of a Parkinson speech dataset with multiple types of sound recordings. *IEEE J. Biomed. Health Inform.* **2013**, *17*, 828–834.
- [7] Drotár, P.; Mekyska, J.; Rektorová, I.; Masarová, L.; Smékal, Z.; Faundez-Zanuy, M. Evaluation of handwriting kinematics and pressure for differential diagnosis of Parkinson's disease. *Artif. Intell. Med.* **2016**, *67*, 39–46.
- [8] Benba, A.; Jilbab, A.; Hammouch, A.; Sandabad, S. Voiceprints analysis using MFCC and SVM for detecting patients with Parkinson's disease. In *Proceedings of the IEEE 2015 International Conference on Electrical and Information Technologies (ICEIT)*, Marrakech, Morocco, 25–27 March **2015**; pp. 300–304.
- [9] Choi, H.; Ha, S.; Im, H.J.; Paek, S.H.; Lee, D.S. Refining diagnosis of Parkinson's disease with deep learning-based interpretation of dopamine transporter imaging. *NeuroImage Clin.* **2017**, *16*, 586–594.
- [10] Haq, A.U.; Li, J.P.; Memon, M.H.; Malik, A.; Ahmad, T.; Ali, A.; Nazir, S.; Ahad, I.; Shahid, M. Feature Selection Based on L1-Norm Support Vector Machine and Effective Recognition System for Parkinson's Disease Using Voice Recordings. *IEEE Access* **2019**, *7*, 37718–37734.

-
- [11] Vasquez-Correa, J.C.; Arias-Vergara, T.; Orozco-Arroyave, J.R.; Eskofier, B.; Klucken, J.; Noth, E. Multimodal assessment of Parkinson's disease: A deep learning approach. *IEEE J. Biomed. Health Inform.* 2019, 23, 1618–1630.
- [12] Avuçlu, E.; Elen, A. Evaluation of train and test performance of machine learning algorithms and Parkinson diagnosis with statistical measurements. *Med. Biol. Eng. Comput.* 2020, 58, 2775–2788.
- [13] Ozsahin, I.; Sekeroglu, B.; Pwavodi, P.C.; Mok, G.S.P. High-accuracy automated diagnosis of Parkinson's disease. *Curr. Med. Imaging Former. Curr. Med. Imaging Rev.* 2020, 16, 688–694.
- [14] - Ding, S.; Zhao, H.; Zhang, Y.; Xu, X.; Nie, R. Extreme learning machine: Algorithm, theory and applications. *Artif. Intell. Rev.* 2015, 44, 103–115.
- [15]-Choi, H.; Ha, S.; Im, H.J.; Paek, S.H.; Lee, D.S. Refining diagnosis of Parkinson's disease with deep learning-based interpretation of dopamine transporter imaging. *NeuroImage Clin.* 2017, 16, 586–594.
- [16] - Qiu, J.; Wang, B.; Zhou, C. Forecasting stock prices with long-short term memory neural network based on attention mechanism. *PLoS ONE* 2020, 15, e0227222.
- [17] -Haralick, Robert M., K. Shanmugam et Its'Hak Dinstein. 1973. « Textural Features for Image Classification ». *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 3, no 6, p. 610-621.
- [18] -O. Régner ' Méthodes d'analyse de texture pour la cartographie d'occupations du sol par télédétection très haute résolution : application au forêt, la vigne et les parcs ostréicoles ' université de Bordeaux 2014)
- [19] -Ojala, Timo, Matti Pietikäinen et David Harwood. 1996. « A comparative study of texture measures with classification based on featured distributions ». *Pattern Recognition*, vol. 29, no1, p. 51-59.
- [20]-T. Ojala, M. Pietkäinen and T MäenpääHarwood (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transaction on Pattern Analysis Intelligence*, 24(7): 971-987.
- [21]-Guo, Z., LeiZhang et DavidZhang (2009). Rotation invariant texture classification using lbp variance (lbpv) withglobalmatching. *Pattern Recognition*.
- [22]- J. Holland, "Adaptation in natural and artificial systems", Second Edition, MIT Press, 1992.Th. Back, D.B. Fogel and Z. Michalewicz. "Handbook of Evolutionary Computation", Oxford University Press (1997).

-
- [23] - :M. Husken ., J. E Gayko, and B. Sendhoff (2000). Optimization for problem classes – neural networks that learn to learn». In X. Yao, editor, IEEE Evolutionary Computation and Neural Networks.IEEE Press, pp98-109.
- [24]- D. Ritchie, C. White, S. Parker, W. Hahn and H. Moore (2003). Optimization of neural network architecture using genetic programming improves detection and modeling of gene-gene interactions in studies of human diseases», BMC Bioinformatics, pp 4:28.
- [25] - A.Berro (2001). Optimisation multi-objectif et stratégies d'évolution en environnement dynamique.Thèse de doctorat, Université des Sciences Sociales.
- [26] - L.J. Eshelman and J.D. Schaffer. "Real-coded Genetic Algorithms and Interval Schemata", In Foundations of Genetic Algorithms - 2, pp. 187-202, Morgan Kaufmann (1993).
- [27] -C.Z. Janikow and Z. Michalewiz. "An Experimental Comparaison of Binary and Floating Point Representations in Genetic Algorithms", In Proceedings of the 4th International Conference on Genetic Algorithms, pp. 31-36, (1991).
- [28] - Mirjalili, S., Mirjalili, S. M., & Lewis, A. (2014). Grey wolf optimizer. Advances in engineering software, 69, 46-61.
- [29]- Heidari, A. A., Mirjalili, S., Faris, H., Aljarah, I., Mafarja, M., & Chen, H. (2019). Harris hawk's optimization: Algorithm and applications. Future generation computer systems, 97, 849.
- [30] - D. Nicolas, – Optimisation de trajectoires pour la résolution de conflits en route. – Thèse de Doctorat, INPT, 1996.
- [31] - Humphries, NE, Queiroz, N., Dyer, JRM, Pade, NG, Musyl, MK, Schaefer, KM, Fuller, DW, Brunnschweiler, JM, Doyle, TK, Houghton, JDR, Hays, GC, Jones, CS, Noble, LR, Wearmouth, VJ, Southall, EJ et Sims, DW (2010). Le contexte environnemental explique les schémas de déplacement Lévy et Browniens des prédateurs marins. Nature, 465(7301), 1066.
- [32] - Bartumeus, F., Catalan, J., Fulco, UL, Lyra, ML et Viswanathan, GM (2002). Optimiser le taux de rencontre dans les interactions biologiques : Lévy versus stratégies browniennes. Lettres d'examen physique, 88(9), 097901.