Introduction

on s’intéresse initialement au problème des centre initiaux de l’algorithme FCM pour ce fait et comme expliquer au chapitre 1 on va utiliser des metaheurstique pour une solution approchée, dans ce chapitre on va parler de l’éventuelle solution de ces mataheristique et leur adaptation pour une rentabilité plus efficace

## II.2. Representation de L’environnement

L’algorithme des C-moyennes floues,établit une optimisation itérative en estimant de façon approximative les minimums d’une fonction d'erreur en utilisant l’indice de flou "fuzzy index" et que détermine le degré de flou de la partition reculée.

La méthode de l'algorithme FCM est basée sur la minimisation d’un critère ou d’une fonction objectif où les clusters sont représentés par les centres respectifs En raison des modèles de données X={x1,x2,...,xN}, l'algorithme de classification de FCM permet d'arrêter la répartition dans l'espace, par calcul dans les centres de classes et la matrice d'appartenance .

# FCM appliqué a la segmentation d’images

Un algorithme de segmentation par classification floue de pixels tels que FCM est composé de deux étapes :

**1.** La division de l’ensemble des pixels en un nombre donné de classes non-disjointes. Durant cette première partie, on cherche à conserver le plus longtemps possible les informations utiles, jusqu’au moment de la prise de décision (*défuzzification*).

**2.** La définition des différentes régions contenues dans l’image, et cela s’effectue par seuillage c’est a dire en choisissant d’attribuer chaque pixel à la classe pour laquelle son degré d’appartenance est maximal. Comme dans les techniques de segmentation par classification il n’est pas essentiel de classer tous les pixels d’une image, mais plutôt les différentes valeurs de niveaux de gris que l’on trouve dans celle-ci et puis l’algorithme FCM peut être adapté au cas de la classification floue des *N* niveaux de gris d’une image

# Les Limitations de FCM

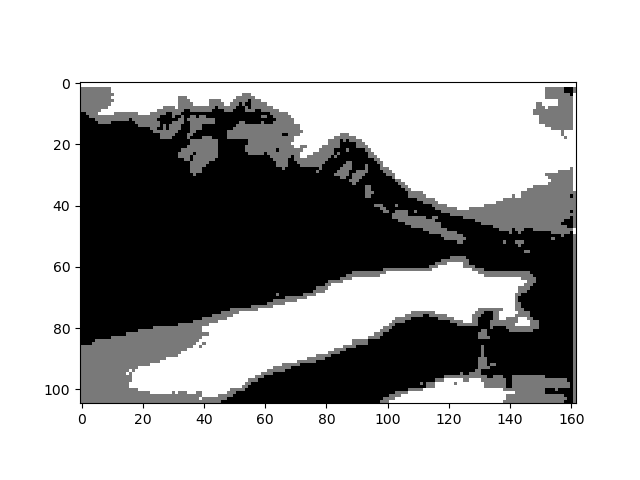
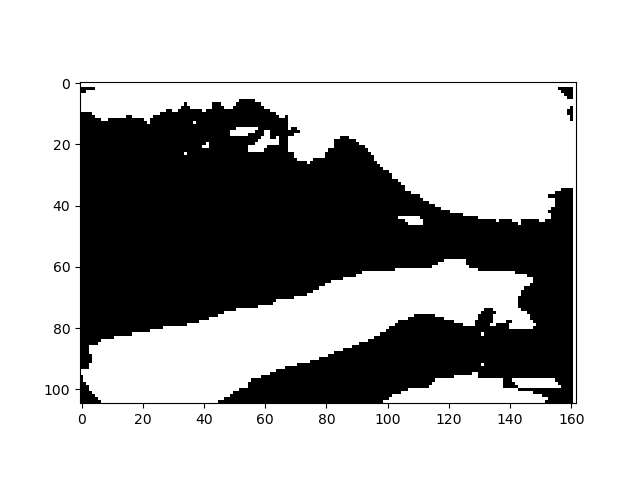
La plupart des méthodes de classification , y compris le FCM, soufrent pratiquement des mêmes défauts

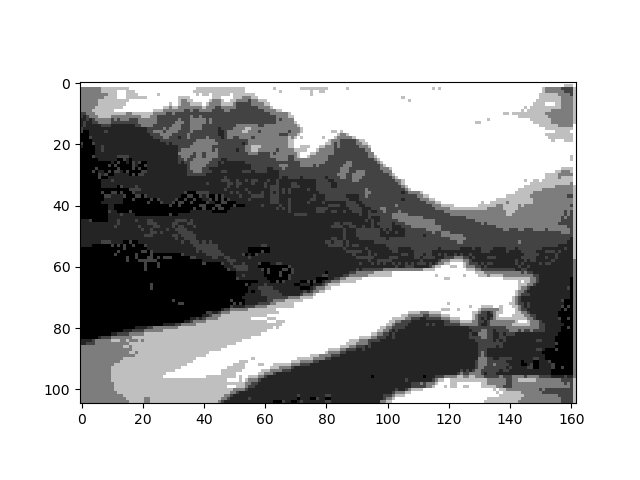
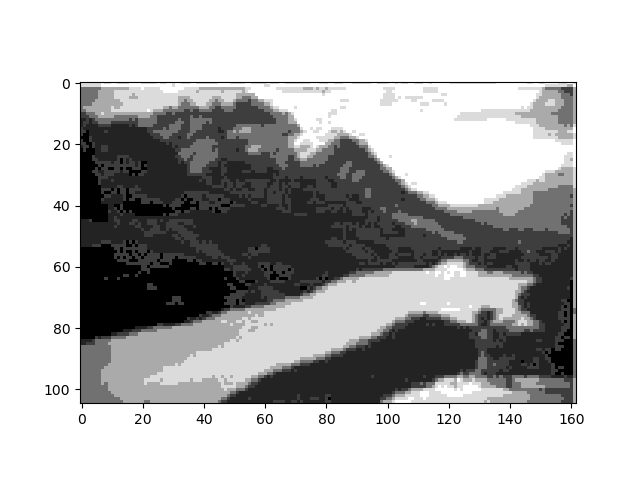
* sensibilité au bruit , une valeur absurde peut changer d’une façon immense

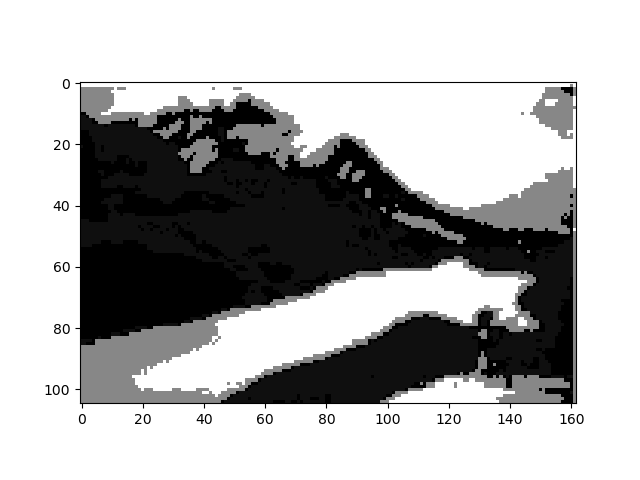
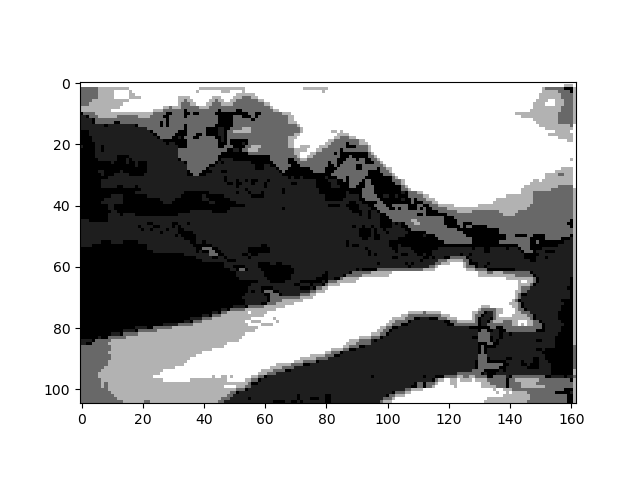
la distribution des classes.

* Le nombre de clusters doit être fourni à l'avance (**Algorithmes supervisés**)
* Chaque région est caractérisée par les degrés d'appartenance qui sont calculés en faisant intervenir la distance euclidienne d'où une forme nécessairement **hyper-sphérique.**
* Le choix des centres de clusters initiaux impact le résultat obtenu, c’est-à-dire que l’initialisation aléatoire peut mener à un blocage dans un minima locale et dans notre projet on s’intéresse a ce dernier

# Exemple sur FCM :

****

****

****

**Segmentation de l’image en utilisant FCM avec C dans [2,7]**

# Contribution des metaheuristique

Pour éviter le faite que l’algorithme FCM peut être très coincé dans les minima locaux, la solution pas nécessairement la plus efficace, est d'utiliser l'algorithme plusieurs fois et commencer chaque fois avec une solution initiale différente, puis de choisir la meilleure, Pour résoudre ce problème nous utilisons les algorithmes d'optimisation d’où les métaheuristique. L'avantage des métaheuristique dans leur consistance est de résoudre les problèmes difficiles pour lesquels les données sont incertaines et bruyantes La solution apportée par la métaheuristique est généralement une solution sous-optimale, souvent très proche de la solution optimale donc on les utilise pour trouver la position optimale des centres de clusters Afin d'optimiser les initialisations de l'algorithme de clustering FCM

## Optimisation de la segmentation par l’utilisation d’histogramme

Pour extraite la distance qui sépare les niveaux de gris de chaque pixel des différents centres, nous utiliserons la distance euclidienne, dans ce cas on peut exploiter la version du FCM basée sur l’histogramme au lieu d’utiliser la version standard (appliquer sur toute l’image) dans ce cas le Coefficient J qui est la mesure a minimiser est donnée par l’équation suivante :

Avec cette optimisation en utilisant l’histogramme deux solutions s’amène a nous la première consiste a appliquer FCM en utilisant les composants de la solution (les centres des classes), puis calculer le coefficient J en se basant sur les centres mis à jour. Quant à la seconde, elle consiste à exécuter une seule itération du processus de FCM et puis calculer le coefficient J, puis n’appliquer tout le processus de FCM que sur la solution obtenue par la métaheuristique

Dans ce cas l’équation qui sert a calculer le degré d’appartenance d’un point j à un centre i est donnée par l’équation

L’équation qui sert à mettre à jour un centre i est donnée

## L’adaptation Des Métaheuristiques

L’ adaptation des différentes métaheuristiques pour la segmentation nous permet de les comparer sur les plans suivants : la complexité, la vitesse de convergence, l’adaptabilité, et la rentabilité des solutions.

## Adaptation de PSO

Cette méthode d'optimisation des particules a l'avantage d'être compatible avec la méthode de regroupement FCM . En fait, les deux algorithmes ont été conçus pour résoudre des problèmes à variables continues, Le problème est abordé en tenant compte de la population (particules), où chaque particule est une solution potentielle au problème.

En premier lieu Les positions initiales et les vitesses des sont choisis de manière aléatoire.

Puis a chaque génération t, la "fitness" f de chaque position xi (t) est calculée, et si g désigne la position du meilleur, pour chaque particule i, la position de chaque particule est mise à jour à chaque étape en fonction de sa meilleure position personnelle et de la meilleure solution du groupes, L’évolution de la solution est assurée par ces deux équations:

X : est la position de la particule (centre d’un segment)

V :est la vitesse de la particule

pbeste la meilleure position de la particule,

gbest la meilleure position globale des particules,

rand1 et rand2 sont des valeurs comprises entre 0 et 1,

c1 et c2 sont des constantes positives qui déterminent respectivement l'impact de la meilleure solution personnelle et de la meilleure solution globale sur le processus de recherche,

k nombre d’itérations

en ce qui concerne la condition d’arrêt on a deux critère si l’une d’entre elles est vérifié l’algorithme s’arrête :

1. l'amélioration non significative de la fonction objective après nerp itérations :

∀k ∈ [0, nbmax],∀n ∈ [k,k +nerp] : J (n+1) - J (n)≺ ep

2.le nombre maximal de répétitions.

## Adaptation de BAT

Afin d'appliquer l'algorithme de chauves-souris à la classification d'images, nous nous sommes inspirés de l'adaptation de l'optimisation par essaim particulaire à la classification Ainsi, nous associons à chaque chauve-souris , un vecteur de dimension d\*k .k étant le nombre de groupes, ou de régions présentes dans l'image.

Pour commencer faut Initialiser la population des chauves-souris, r et A aléatoirement puis.

Pour (t < nombre maximale d’itérations) :

Pour chacune des chauves-souris Générer des nouvelles solutions par ajustement des fréquences et mise à jour des vélocités et cela avec les equations suivant :

Si ( rand > bat.r) alors

Fitness = J(S)

Si ((Bat.fitness > Fitness) et (rand < bat.A ) )alors

Si ( Fitness > best.fitness ) alors

S [ ]

Retourner Best

# Adaptation de GOA :

la principale caractéristique de la sauterelle au premier temps est de se déplacer lentement et de se nourrir à petite échelle. Lorsqu'elles deviennent adultes, l'action collective devient la principale caractéristique de l'activité de la sauterelle, comme on le sait Les algorithmes inspirés de la nature divisent le processus de recherche en deux tendances : l'exploration et l’exploitation. En matière d'exploitation, les agents de recherche sont encouragés à se déplacer brusquement, alors qu'ils ont tendance à se déplacer localement pendant l'exploitation. Ces deux fonctions, ainsi que la recherche de cibles, sont assurées par des sauterelles naturelles. Par conséquent, on modélisant ce comportement mathématique des sauterelles traditionalistes on aura :

Xi=Si +Gi +Ai

Où Xi définit la position de la i-ième sauterelle, Si est l'interaction sociale, Gi est la force de gravité sur la i-ième sauterelle, et Ai indique le mouvement du vent.

où dij est la distance entre la i-ème et la j-ème sauterelle, calculée comme dij=|xj-xi| , s est une fonction pour définir la force des forces sociales, d⃗ ij est un vecteur unitaire de la i-ème sauterelle à la j-ème sauterelle.

où g est la constante gravitationnelle et eg montre un vecteur unité vers le centre de la terre.

u est une dérive constante et ew est un vecteur unité dans la direction du vent.

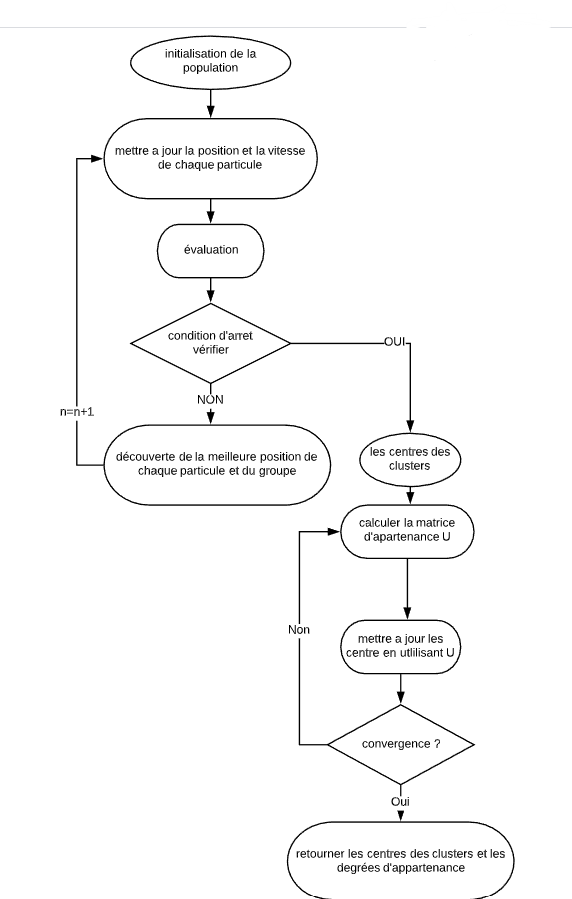
En remplaçant S, G et cette équation peut être développée comme suit :

+

Ce modèle mathématique ne peut toutefois pas être utilisé directement pour résoudre des problèmes d'optimisation, principalement parce que les sauterelles atteignent rapidement la zone de confort et que l'essaim ne converge pas vers un point précis. Une version modifiée pour résoudre les problèmes d'optimisation :

Parmi celles-ci, ubd et lbd sont un type de limitation supérieure et inférieure, T→d est la valeur optimale après chaque itération, c1=c2=cmax-lcmax-cminL,c1 équilibre la recherche globale et la recherche locale pour la zone cible, c2 équilibre la relation entre l'attraction entre deux sauterelles, cmax et cmin peuvent fixer la capacité de recherche maximale et minimale, l représente le numéro d'itération actuel, L est le plus grand nombre d'itérations.

## Organigramme générale :



# Récapitulatif générale :

On a vu que l’application de l’algorithme FCM pour la segmentation d’image comporte quelque problèmes qui peuvent mener vers des résultats erronés pour ce fait on a utilisé des méthodes approchés tel que les Metaheuristique et comme Metaheuristique on a utilisé la (particules swarms optimization ) PSO , BAT algorithm for optimization , GOA (grasshopper optimization algorithm) et les sont des meta a population de solution c’est-à-dire on lance les particules et on calcule leur meilleure position par rapport a leur position initial et celle du groupe avec des contrainte tel que la vitesse , la fréquence pour BAT , point d’attraction pour GOA ou chaque particule donc sa position est une éventuelle solution donc les position produite seront considérer comme les centres initiaux des clusters au lieu d’une initialisation aléatoire pour l’application de FCM par la suite on va calculer la matrice d’appartenance en minimisant la fonction objective respectivement avec ces centres et les dégrées d’appartenance.

# Implémentation :

## Codage :

Pour une image contenant N régions, une solution est indiquée par un tableau de taille N ou le *i*è*me* élément représente la valeur du centre de la *i*è*me* classe, vérifiant la condition suivante :

*Min≤*  *Solution*[*i*] ≤*Max* [1;*N*]

Où *Min* et *Max* font référence respectivement à la plus petite et à la plus grande valeur de niveau de gris disponible dans l’histogramme de niveau de gris de l’image à segmenter.

# Architecture utilisée :

## La bibliothèque Deap de python :

DEAP (Distributed Evolutionary Algorithms in Python) son objectif est de fournir une boîte à outils qui encourage les utilisateurs d'écrire leurs propres algorithmes évolutifs, en contrôlant explicitement chaque aspect du processus évolutif tel que les types de données, mesures d'aptitude, initialisation de la population, opérateurs, boucle d'évolution, etc. en gros avoir des boucles d'évolution où chaque étape est explicite, et courtes, faciles à lire et à comprendre, et facile à documenter , Un autre objectif fondamental de DEAP est de fournir un parallélisme transparent, autant que possible. Les chercheurs veulent expérimenter de nouvelles idées pour résoudre des problèmes complexes du monde réel. Leur préoccupation première est de valider dans un délai raisonnable si ces Les idées sont bonnes. Avec la stagnation de l'horloge du processeur et la complexité habituellement élevée des problèmes du monde réel, le parallélisme est donc une exigence essentielle

# Conclusion :

Dans ce chapitre on a discuté la solution pour la problématique de l’initialisation de l’algorithme de segmentation FCM tout en abordant l’adaptation des différents Metaheuristique dans le prochain chapitre on pourra étudiée en profondeur ces méthodes approchées tout en comparant les résultats.