Dernière mise à jour	Informatique	Denis DEFAUCHY – <u>Site web</u>
26/01/2023	3 – Intelligence artificielle	TD 3-4 – KNN 2D

Informatique

3 Intelligence artificielle

TD3-4 KNN 2D



Dernière mise à jour	Informatique	Denis DEFAUCHY – <u>Site web</u>
26/01/2023	3 – Intelligence artificielle	TD 3-4 – KNN 2D

Contexte

A partir de points 2D répartis par paquets/familles, chaque paquet étant positionné aléatoirement sur une image, on souhaite créer une nouvelle image dont chaque pixel possède la couleur de la famille à laquelle il sera affecté par application de l'algorithme KNN pour différentes valeurs de k.



Création de données

A partir des définitions suivantes :

- Nl=100 : Nombre de lignes de pixels de l'image
- Nc=100 : Nombre de colonnes de pixels de l'image
- Ng=5 : Nombre de groupes de points
- Nppg=10 : Nombre de points par groupe
- dlc=5: Les points de chaque groupe doivent être répartis aléatoirement dans des carrés de côté 2dlc+1

On souhaite avoir à disposition les éléments suivants :

- LP: Liste de tous les points, chacun étant une liste à deux termes [L,C] indiquant ses ligne et colonne dans les limites de l'image
- LG: Liste des Ng groupes, chaque groupe G étant un ensemble de points associé à une famille
- LNG: Liste des numéros de groupes telle que LNG[i] est le numéro du groupe du point LP[i]
- LC : Liste de couleurs aléatoires telle que LC[i] est la couleur du groupe LG[i]. Une couleur est un triplet [R,G,B] avec R,G et B des nombres entiers entre 0 et 255

Question 1: Mettre en place le codé créant ces éléments

Vérifier que vous obtenez des résultats ressemblant à ceux-ci :



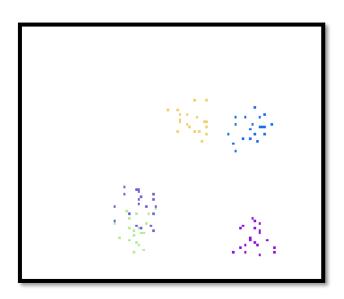
Dernière mise à jour	Informatique	Denis DEFAUCHY – <u>Site web</u>
26/01/2023	3 – Intelligence artificielle	TD 3-4 – KNN 2D

Initialisation de l'image

Question 2: Proposer une fonction Affiche(fig,im) affichant une image (array) sur la figure numéro fig

Question 3: Mettre en place les lignes de code créant une image blanche aux dimensions NI,Nc, affectant la couleur voulue à chaque pixel des données, et affichant cette image sur la figure 0

Exemple:



Mise en place de l'algorithme KNN

Si vous avez déjà fait le TD3-1 sur la reconnaissance automatique de panneaux, allez récupérer ces 4 fonctions que vous avez déjà programmées.

On rappelle que la distance euclidienne entre deux n-uplets $u=(u_0,u_1,\ldots,u_{n-1})$ et $v=(v_0,v_1,\ldots,v_{n-1})$ est le résultat du calcul suivant :

$$D = \sqrt{\sum_{i=0}^{n-1} (v_i - u_i)^2}$$

Question 4: Créer une fonction Distance_uv(u,v) calculant la distance euclidienne entre les deux n-uplets sous forme de listes u et v

Vérifier:



Dernière mise à jour	Informatique	Denis DEFAUCHY – <u>Site web</u>
26/01/2023	3 – Intelligence artificielle	TD 3-4 – KNN 2D

Question 5: Créer une fonction Distance(u,Lv) renvoyant une liste des distances euclidiennes entre u et tous les n-uplets v de la liste Lv ainsi que l'indice associé sous la forme [Distance entre u et v,indice de v dans Lv]

```
Vérifier: >>> u = [1,3,5]
>>> v1 = [1,2,3]
>>> v2 = [2,4,6]
>>> v3 = [1,3,5]
>>> Lv = [v1,v2,v3]
>>> Distance(u,Lv)
[[2.23606797749979, 0], [1.7320508075688772, 1], [0.0, 2]]
```

Question 6: Créer la fonction Proches(u,Lv,k) qui renvoie une liste des k plus proches voisins de u dans la liste Lv au sens de la norme euclidienne, soit les k listes [dst,ind]

Remarques:

- On supposera que k est plus petit que le nombre de n-uplets de Lv
- On autorise l'utilisation de L.sort() pour trier les éléments de L en place par rapport à la première composante de tous ses éléments (les distances ici)

```
A la suite du code précédent, vérifier : >>> Proches (u, Lv, 1) [[0.0, 2]]
```

Question 7: Créer une fonction Max_Occurences(L,n) qui renvoie le terme apparaissant le plus dans L, et le premier s'il y a des exæquo avec n nombre de dossiers

Identification de la famille d'appartenance

Pour la suite, les données LP, LNG et LC, NI, Nc et Ng seront utilisées de manière globale, on ne le mettra donc pas en arguments des différentes fonctions créées.

On souhaite déterminer la famille d'appartenance d'un pixel quelconque de l'image Pix=[lp,cp] à l'aide de la méthode knn.

Question 8: Proposer une fonction Famille(Pix,k) renvoyant la famille d'appartenance (indice dans LNG)du pixel Pix en étudiant ses k plus proches voisins parmi les points de LP



Dernière mise à jour	Informatique	Denis DEFAUCHY – <u>Site web</u>
26/01/2023	3 – Intelligence artificielle	TD 3-4 – KNN 2D

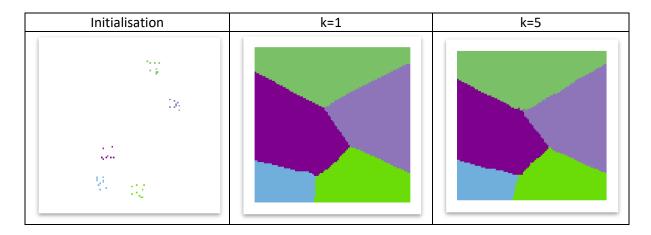
Application à l'image

Nous allons maintenant changer les couleurs des pixels de toute l'image initiale afin d'obtenir le résultat souhaité.

Question 9: Créer la fonction Couleur_Pix(Pix,k,im) déterminant la famille à laquelle appartient le pixel Pix avec k plus proches voisins, et affectant la couleur de cette famille à ce pixel dans l'image im

Question 10: Créer la fonction Image_k(k) créant une copie de l'image image initialisée en début de sujet, et réalisant l'ensemble des modifications de tous ses pixels

Question 11: Utiliser Image_k pour créer les images de numéro k pour différentes valeurs de k



Vous pourrez maintenant changer les paramètres initiaux afin d'obtenir de plus belles images :

