

# TP6

# IAA

**Classification et évaluation**

## ***1. Classification non supervisée***

Le calcul des composantes TLS (HSV en anglais) mets un certain temps. J'ai réadapté l'algorithme des kmeans du TP1 au cas présent. Les centres (ou les teintes) initiaux sont pris aléatoirement en faisant attention de ne pas prendre deux centre identiques.

Le critère d'arrêt ne porte pas sur l'inertie mais sur le fait qu'on retombe sur les même centres après une itération, ce qui fait que le nombre d'itération peut être important.

Les distances entre teintes sont calculés en faisant une soustraction (i.e la teinte 0.1 est aussi éloigné de 0.2 que 0.4 l'est de 0.5, mais moins que 0.3 l'est de 0.5), une distance euclidienne en dimension 1 donc.

Les fichiers concernés dans ce tp sont tp6.py et kmeansTeinte.py. Assurez-vous qu'ils sont dans le même dossier ainsi que l'image que vous voulez traiter.

Pour appliquer l'algo sur d'autres images avec différentes classes il faut modifier le fichier tp6.py

Mettez l'emplacement de l'image à traiter (l.14), spécifier le nombre K de classes désirées (l.37), et spécifiez l'emplacement de sauvegarde de la nouvelle image (l.72).

### **Observation des résultats :**

Pour évaluer la classification produite, on peut changer la valeur de la teinte pour chaque pixel par la moyenne de la classe à laquelle appartient la teinte. On obtient les résultats visuels suivant :



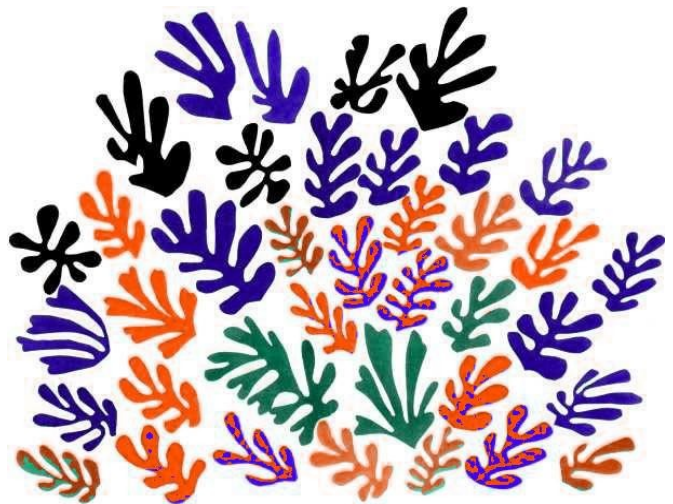
*Illustration 1: K-means 1 teinte appliqué sur Matisse*



*Illustration 2: K-Means 2 teintes appliqués sur Matisse*



*Illustration 3: un exemple de K-means 3 teintes*



*Illustration 4: un autre exemple pour K-means 3 teintes*

Ci dessus quelques exemples de K-means utilisés sur l'image Matisse. On peut avoir des résultats différents pour une même valeur de K (illustration 3 et 4) car les centres initiaux sont choisis aléatoirement.

On observe qu'il y a visuellement quelques erreurs de classification, illustrations 2,3 et 4 certaines feuilles ont 2 teintes différentes formant des sortes de tâches alors qu'elles devraient avoir une teinte uniforme.



*Illustration 5: Matisse image original*

Voici ci-dessus l'image de départ, il s'agit de la dernière œuvre de Matisse, « *La Gerbe* » tableau peint en 1953.

Il faut environ 20 teintes pour obtenir une image similaire. Certaines teintes sont en effet très proche tout en étant différentes.



*Illustration 6: Matisse avec 20 teintes pour K-means*

Pour le tableau de Joan Mirró, qui est une lithographie non signée datant de 1956, au bout de 5 teintes on a déjà une image proche de l'original :





*Illustration 7: miro 1 teinte*



*Illustration 8: Miro 2 teintes*



*Illustration 9: Miro 3 teintes*



*Illustration 10: Miro 4 teintes*



*Illustration 11: Miro 5 teintes*



*Illustration 12: Miro original*

Ici on observe des résultats intéressants : Avec une teinte, on a que du vert. Avec 2 teintes, on perd le vert et on gagne bleu et orange. A 3 teintes on regagne le vert, à 4 teintes le vert et le bleu sont remplacés par le « bleu vert » mais on gagne le rouge et le jaune. A 5 teintes on regagne le vert et le bleu. On en déduit que le choix de K est important et qu'en augmentant K on obtient pas forcément les mêmes teintes.



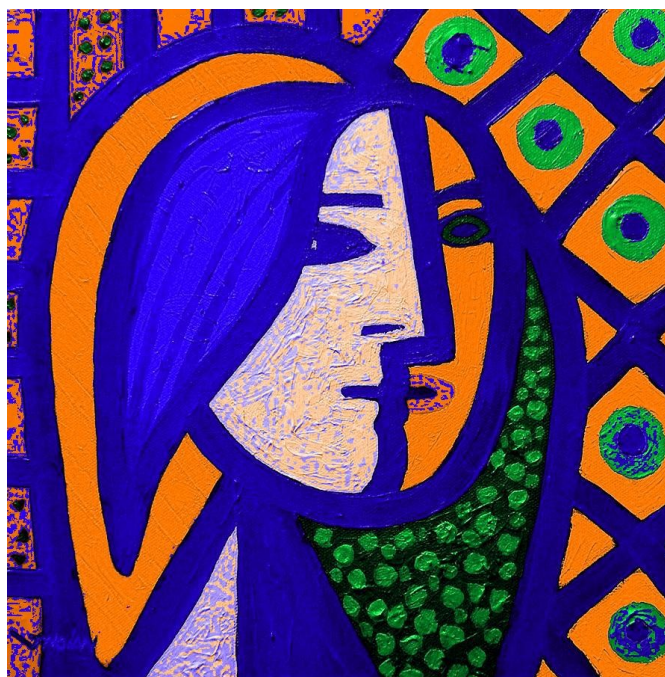
Le tableau de Picasso est aussi une œuvre sans nom. Voici les résultats observés :



*Illustration 13: picasso 1 teinte*



*Illustration 14: Picasso 2 teintes*



*Illustration 15: Picasso 3 teintes*



*Illustration 16: Picasso 4 teintes*





*Illustration 17: Picasso 5 teintes*



*Illustration 18: Picasso original*

Ici on observe le même phénomène, la teinte jaune présente pour le cas 2 teintes ne réapparaît qu'au cas 5 teintes. Avec 5 teintes on commence déjà à avoir une image similaire. Les ronds à l'arrière plan ont cependant la même teinte alors qu'ils devraient être séparés en 2 teintes.

Pour évaluer la performance de la segmentation de manière automatique, il serait possible de mettre en place une mesure de corrélation entre les images obtenues et les images originales. Mais là aussi il y a plusieurs possibilités, fait-on une comparaison dans RGB ? Dans HSV ? Que signifie comparer deux images ? Tout dépend de ce que l'on cherche à obtenir.

On peut aussi utiliser des notions d'inerties vu dans les Tps précédents et en cours pour évaluer la qualité de la segmentation.

## **Contact**

Si il y a des erreurs, des remarques, des ajouts à faire, etc.

Veuillez m'en faire part à cette adresse :

wedg@hotmail.fr