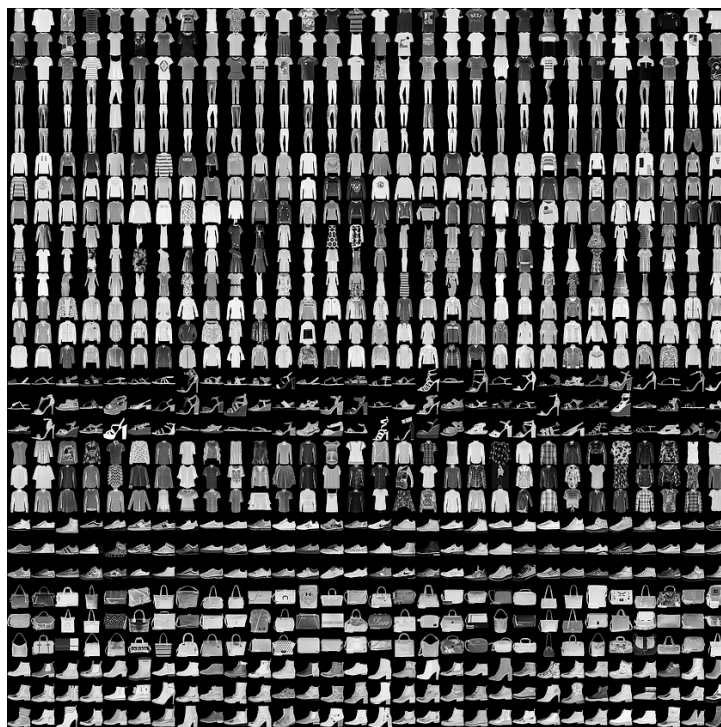


TD de Vision par ordinateur

Partie 4 : Reconnaissance de formes

Dans ce TP, nous allons implémenter des algorithmes de reconnaissance de forme sur les bases de données 'fashion.png' contenant des images de vêtements et 'digits.png' contenant des images de digits manuscrits. Nous allons utiliser deux codages pour les formes à reconnaître : le codage rétinien puis le codage HOG (histogrammes de l'orientation du gradient). Puis nous utiliserons l'algorithme des k-plus proches voisins dans le cas supervisé le l'algorithme k-means pour le cas non supervisé.



1/ Reconnaissance par apprentissage supervisé :

Les bases utilisées sont implicitement étiquetées car les imagerie sont ordonnées de la première à la dixième classe. Nous allons diviser la base de données en une partie entraînement et une partie test.

a/ Codage rétinien : l'image 'digits.png' contient 5000 imagerie des digits de 0 à 9 manuscrits. Chaque imagerie fait 20x20 pixels. Charger cette image puis créer la matrice 'exemples_entraînement' dont chaque ligne est un vecteur code d'une imagerie d'ordre impair. Faites de même pour créer la matrice 'exemples_test' pour les imagerie d'ordre pair.

b/ Implémentez l'algorithme des k-plus proches voisins pour reconnaître une image choisie dans la base de test. Faire varier k de 1 à 10. Conclusion.

d/ Ecrire un script permettant d'évaluer le taux de réussite de votre algorithme.

c/ Effectuer les mêmes étapes sur la base 'fashion.png'. Cette base contient 900 images de vêtements de 10 catégories. Attention, les imagerie sont cette fois-ci de taille 28x28 pixels.

d/ Refaire les même étapes en remplaçant le codage rétinien par un codage HOG. Ce dernier consiste à diviser l'imagerie en $m \times m$ block et de calculer l'histogramme des orientations du gradient dans chaque block sur 12 barres (30° par barre). Faire varier m et k .

e/ Comparer les taux de succès et conclure.

2/ Apprentissage non supervisé :

a/ Implémenter l'algorithme des k-means sur chacune des bases d'entraînement précédentes et avec chaque codage en partant de 10 classes prototypes choisies de façon aléatoire.

b/ Afin de vérifier que la classification s'est bien passée, refaire l'estimation des taux de succès avec les bases de test.