

TP3

Traitement de données – Classification automatique d'images (SVM et MLP)

L'objectif est de concevoir un système capable de classer automatiquement des images en différentes catégories. On dispose pour cela d'une banque de 2288 images numérotées. Chaque image possède 1 catégorie dite principale et 1 catégorie dite secondaire ou sous-catégorie.

La première catégorie représente la classe principale de l'image, elle peut être :

- Artificielle (constructions humaines)
- Naturelle (image de nature)

La seconde catégorie représente la sous-classe de l'image, il y en a 8 :

- Côte (plage, mer...)
- Forêt
- Autoroute
- Ville (Photo prise de loin de l'ensemble de la ville)
- Montagne
- Paysage ouvert (landes, désert...)
- Rue
- Grand bâtiment

Chaque image est décrite à l'aide de 27 vecteurs de caractéristiques (features en anglais). Les 24 premiers vecteurs de caractéristiques sont issus d'un banc de filtres de Gabor et consistent à récupérer l'énergie dans chacune des bandes de fréquences (ici 24). Les 3 derniers vecteurs de caractéristiques sont issus de la luminance Y et de la chrominance Cb puis Cr.

Pour information, les images sont accessibles dans le fichier images.zip et peuvent être visualisées.

L'objectif de ce projet est de proposer un classifieur parmi un SVM et un MLP qui permette de répondre le mieux au problème de classification posé (classification images naturelles/artificielles, classification par catégories d'images).

Pour cela, commencer par partitionner la base de données en deux bases : apprentissage et test. Visualiser les données de la base d'apprentissage en réalisant une ACP et une ALD et en projetant les données sur les 2 premiers axes.

Ensuite, pour les 2 classifieurs (MLP et SVM), choisir la structure la plus adaptée au problème de classification. On utilisera comme critère de sélection une métrique calculée par validation croisée sur la base d'apprentissage.

Une fois la structure des classifieurs choisie, évaluer et comparer leurs performances sur la base de test, en prenant soin de calculer les intervalles de confiance des métriques de performances choisies.

Conclure sur le choix des classifieurs retenus.

Vous pourrez évaluer la sensibilité des résultats à la taille de la base d'apprentissage. Vous pourrez aussi évaluer la perte en performances quand on supprime les 3 dernières caractéristiques, c'est-à-dire la luminance Y, la chrominance Cb et Cr.

Avant de programmer un MLP ou un SVM, il est nécessaire d'étudier sur le site de sikitlearn la documentation relative à ces classifieurs : MLPClassifier et svc, ainsi que la fonction permettant la validation croisée, cross_validate . Il est nécessaire de prendre le temps de comprendre les paramètres d'entrées et les attributs associés à chaque fonction.

https://scikit-learn.org/stable/modules/neural_networks_supervised.html#multi-layer-perceptron
https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html

<https://scikit-learn.org/stable/modules/svm.html>
<https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html>

Pour le MLP et le SVM regarder bien la paragraphe « Tips on Practical Use »

https://scikit-learn.org/stable/modules/cross_validation.html
https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.cross_validate.html#sklearn.model_selection.cross_validate

Pour le MLP, on fixera le nombre de couches cachées à 1. On optimisera le nombre de neurones dans la couche cachée et on choisira la fonction d'activation associée aux neurones.

Pour le SVM, on choisira le type de noyau et la constante de pénalité C.

Faites tout le TP pour 2 classes et si vous avez le temps vous pouvez refaire tourner votre code pour 8 classes.