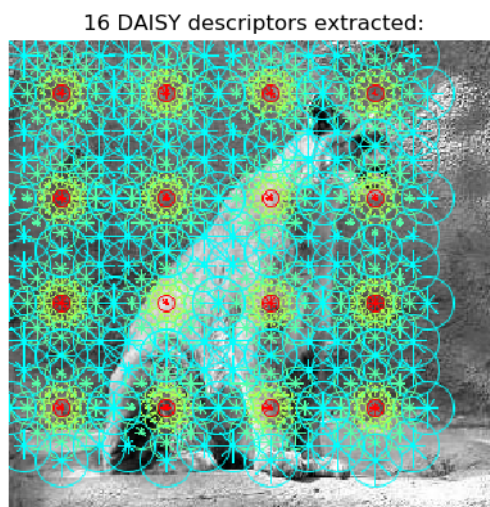


SY32 – TD Apprentissage 03bis : Classification d'images et reconnaissance par la méthode sac de mots visuels

Exercice 1 : Recherche d'images similaires par sac de mots visuels

La fonction `daisy` de `skimage.feature` permet de calculer des descripteurs locaux répartis de manière dense dans l'image. La sortie de `daisy` est une matrice de taille P, Q, R . Les vecteurs de description sont codés selon la dernière dimension. Il y a ainsi $P \times Q$ descripteurs de dimension R . Dans ce TD, nous utilisons un sous ensemble de données du jeu de données Caltech-101 <https://data.caltech.edu/records/mzrjq-6wc02>.



1. Calculer les descripteurs DAISY pour chaque image du dossier `images` (une image donne $P \times Q$ descripteurs) et les concaténer dans une unique matrice de descripteurs (réunir tous les descripteurs de toutes les images dans une seule matrice à deux dimensions, qui servira à alimenter le `KMeans` de la question suivante).
Note : on pourra utiliser les paramètres suivants `step=12`, `radius=20`, `rings=3`, `histograms=8`, `orientations=8`.
2. Utiliser la fonction `KMeans` de `sklearn.cluster` pour calculer un dictionnaire de mots visuels à partir des descripteurs DAISY calculés précédemment.
3. Pour chaque image, calculer un histogramme sur le dictionnaire de mots (c'est-à-dire, compter le nombre d'apparition de chaque mot, servira de descripteur global de l'image).
4. Afficher quelques images différentes et leur histogramme basé sur le dictionnaire de mots. Des images représentant les mêmes éléments devraient avoir des histogrammes semblables.
5. Trouver les dix images les proches de l'image `cougar.jpg` en termes d'histogramme. Faire de même pour les images `carb.jpg` et `kangaroo.jpg`.
6. Maintenant, refaire la recherche des images les plus proches en utilisant des histogrammes normalisés (la somme des valeurs doit être 1).
7. Faites varier les différents paramètres et observer.
8. Refaire l'exercice en utilisant l'extraction et la description de *features* SIFT `skimage.feature.SIFT` ou ORB `skimage.feature.ORB` (SIFT nécessite `scikit-image` version 0.19 ou supérieur).

Note :

Le sac de mots est généralement utilisé pour alimenter des classifieurs type SVM, comme représentation robuste des caractéristiques des images.

Liste de fonctions utiles :

— skimage.color.rgb2gray	— numpy.vstack
— skimage.feature.daisy	— numpy.concatenate
— skimage.feature.ORB	— numpy.squeeze
— skimage.feature.SIFT	— np.expand_dims
— skimage.io.imread avec argument as_gray	— np.argsort
— sklearn.cluster.KMeans	— matplotlib.pyplot.plt.bar
— numpy.bincount avec argument minlength	— scipy.spatial.distance.cdist
— numpy.reshape	— scipy.spatial.distance_matrix