

TP Capteurs-Vision

SEATECH

2^{ème} Année SYSMER et IMECAD

Audrey Minghelli
Durée des TP 3h

Durant les 3 séances de TP, vous allez coder en Python avec la librairie OpenCV ou en Matlab pour manipuler différents outils vus en cours.

Le compte rendu de tous les TP (une seul fichier) est à rendre 2 semaines après la fin des TP au format pdf (avec le code en annexe) sur le site Moodle de l'université.

Tout au long de ces TP vous ferez des copies d'écran que vous stockerez dans un fichier Word pour illustrer votre compte rendu de TP.

TP1 : vision industrielle

Ce TP a pour objectif de faire de comptage automatique d'objets dans une image pour des applications de vision industrielle. Ce TP sera codé en python.

1. Affichage une image

Dans votre script, importer la librairie numpy (`import numpy as np`) destinée à manipuler des matrices ou tableaux multidimensionnels ainsi que des fonctions mathématiques opérant sur ces tableaux.

Importer la librairie OpenCV (`cv2`), qui est une bibliothèque graphique libre, spécialisée dans le traitement d'images en temps réel.

Importer le module pyplot de la librairie matplotlib et renommer `plt` (`import matplotlib.pyplot as plt`). Cette librairie va nous permettre de tracer des courbes.

Ouvrir l'image couleur `test_3bb_blancs.png` (`cv2.imread(...)`)

Afficher le nombre de lignes, colonnes et bandes (`....shape`).

Afficher l'image couleur (`cv2.imshow(...)` puis `cv2.waitKey(0)`).

Ne pas Fermer cette figure à la fin du programme (`cv2.destroyAllWindows()`)

Créer une fonction dans le même programme qui convertit l'image couleur en niveau de gris (en moyennant les 3 bandes) et l'afficher.

2. Histogramme

Créer une fonction dans le même programme qui calcule l'histogramme de l'image en niveau de gris

L'afficher.

```
plt.plot(h)
plt.show()
```

Fermer cette figure à la fin du programme (`plt.close("all")`)

3. Binarisation :

A l'aide de l'histogramme précédent, déterminer un seuil pour la binarisation
Ecrire une fonction qui binarise une image en niveau de gris (mettre 0 sous le seuil, 255 au-dessus)
L'appliquer à l'image précédente
Afficher l'image binarisée et son histogramme

4. Comptage de pixels par bonbon (calibration)

A partir de l'histogramme de l'image binarisée, calculer le nombre de pixels par bonbon sachant que dans cette image il y a 3 bonbons.

5. Comptage de bonbons

Ouvrir une couleur test_7bb_blancs.png (par exemple)

La convertir en niveau de gris

Afficher cette image

Calculer son histogramme

L'afficher

Binariser l'image (avec le bon seuil)

Afficher l'image binarisée

Calculer son histogramme

En déduire le nombre de bonbons dans l'image et l'afficher

Tester le comptage avec toutes les images, expliquer les erreurs s'il y en a.

TP2 : Télédétection par drone

Ce TP a pour objectif de faire manipuler des images multispectrales acquises par un drone sur un champ de betteraves cultivé. 3 images ont été acquises à des temps de croissance différents. Vous utiliserez le langage Matlab pour réaliser ce TP.

1. Analyse d'une image

Charger l'image '425aa_2405.mat'

Quelle est la taille de cette image (size et workspace)

Les bandes des images sont : 450, 530, 560, 675, 730 et 850 nm.
Leur résolution spatiale est comprise entre 2 et 6 cm.

Afficher dans la même figure les 6 bandes spectrales (imshow, subplot). Commenter.

Afficher une composition colorée de l'image en vraies et fausses couleurs.

Afficher dans une même autre figure les histogrammes des 6 bandes. Commenter.

Avec la fonction Impixelinfo, localiser un pixel de sol et un pixel de végétation (attention, x et y inversés)

Tracer sur une même autre figure les profils spectraux (reflectance) d'un pixel de betterave et d'un pixel de sol. Commenter

2. Indice de végétation (NDVI)

Calculer l'indice de végétation de cette image $IV = (PIR - R) / (R + PIR)$

L'afficher avec une table de couleur (colormap ('jet')) et une échelle de couleur (colorbar)

Interpréter l'image.

3. Comparaison avec d'autres dates

Afficher dans une même figure les 3 images en vrai couleurs et en fausse couleur (en mettant du PIR à la place de la bande rouge). Commenter.

Afficher sur une même autre figure les 3 IV avec la même table et échelle de couleur.
Commenter

4. Calcul de la proportion de culture au cours du temps

Appliquer un seuillage à l'indice de végétation pour chaque date.
Calculer et tracer la proportion de culture au cours de temps.

TP3 : Cartographie des fonds par images sous-marines

Ce TP a pour objectif de faire de cartographier des fonds marins par imagerie sous-marine.
Pour ce TP on utilisera le langage Matlab.

1. Affichage de l'image en couleur

Lire l'image « image_fond_ifremer.tif »

Quelle est sa taille ?

L'afficher et la commenter.

Identifier le nombre d'items à cartographier.

Afficher les infos ligne, colonne, R, V, B en dessous de l'image (impixelinfo)

2. Affichage des bandes et leur histogramme

Afficher chaque bande indépendamment (Rouge, vert et Bleu)

Ecrire une fonction qui calcule l'histogramme d'une bande.

Afficher en dessous de chaque bande l'histogramme correspondant.

Commenter

3. Classification supervisée

Sélectionner avec la souris des échantillons (régions d'intérêt ou ROI) pour chaque classe (roipoly).

Calculer la valeur moyenne de chaque échantillon dans les 3 bandes spectrales.

Tracer leur profil (spectral) sur un même graphe.

Appliquer la classification à l'image en associant chaque pixel à la classe la plus proche en testant 2 distances :

- Distance euclidienne

$$d(x, r_k) = \sqrt{\sum_{i=1}^N (x(i) - \bar{r}_k(i))^2}$$

- Spectral Angle Mapper

$$\alpha(x, r_k) = \cos^{-1} \left(\frac{\sum_{i=1}^N x(i) * \bar{r}_k(i)}{\sqrt{(\sum_{i=1}^N x(i)^2)} * \sqrt{(\sum_{i=1}^N \bar{r}_k(i)^2)}} \right)$$

Quelle distance donne les meilleurs résultats ? Expliquer