ISM MPSI/PCSI, Informatique
Année 2025/2026

TP Informatique 25

Récupérer sur le site de la classe le fichier `TP25.py` à compléter, ainsi que les images pour tester.

⚠ Tous ces fichiers doivent être enregistrés dans le même dossier.

La fonction `importer(image)` d'argument `image` un fichier d'image renvoie une liste de listes, chaque sous-liste représentant une ligne de points de l'image où les points sont des valeurs dans $[\![0\,;255]\!]$ qui désignent un niveau de gris (0 : noir, 255 : blanc). La fonction `afficher(tab)` d'argument `tab` une liste de listes représentant une image réalise l'affichage de celle-ci.

Par exemple, avec le fichier `rover.jpg` présent dans le répertoire de travail, on saisit :

```
>>> tab=importer('rover.jpg')
>>> afficher(tab)
```

I Manipulations de base

Exercice 1

Écrire une fonction `negatif(tab)` qui renvoie une liste de listes représentant le négatif de l'image. Pour cela, on inverse le niveau de gris de chaque point en le passant d'une valeur x à sa valeur complémentaire $255 - x$.

Exercice 2

Dans cet exercice, on se propose d'écrire des fonctions permettant d'assombrir/éclaircir une image. On utilise pour cela une nouvelle notion : soient a et b dans un espace vectoriel réel, on appelle *combinaison convexe* de a et b tout élément de la forme $\lambda a + (1-\lambda)b$ avec $\lambda \in [0\,;1]$. Ceci correspond à la notion de *barycentre* utilisé en physique.

Étant donné un point de valeur x et un coefficient $s \in [0\,;1]$, on assombrit le point du facteur s en lui appliquant la combinaison convexe

$$\text{assombrissement} : x \mapsto s \times x + (1-s) \times 0$$

et on l'éclaircit du facteur s en lui appliquant la combinaison convexe

$$\text{éclaircissement} : x \mapsto s \times x + (1-s) \times 255$$

Après transformation, la valeur du point doit être convertie en entier.

1. Écrire une fonction `assombrir(tab,s)` qui renvoie une liste de listes représentant l'image assombrie d'un facteur `s` à valeur dans $[0\,;1]$.
2. Écrire une fonction `eclaircir(tab,s)` qui renvoie une liste de listes représentant l'image éclaircie d'un facteur `s` à valeur dans $[0\,;1]$.
3. Tester les fonctions.

II Transformation géométriques

Pour appliquer des transformations géométriques à une image, on se propose de passer du système de repérage par ligne et colonne (i, j) à un système de coordonnées classiques (x, y). En notant (i_0, j_0) les numéros de ligne et colonne d'un point central de l'image, on obtient les relations suivantes :

$$\begin{cases} x = j - j_0 \\ y = i_0 - i \end{cases} \quad \text{et} \quad \begin{cases} i = \lfloor i_0 - y \rfloor \\ j = \lfloor j_0 + x \rfloor \end{cases} .$$

Exercice 3

Écrire une fonction `homothetie(tab,mu)` qui réalise la dilatation ou rétraction d'une image par un facteur $\mu > 0$ en transformant chaque point de coordonnées (x, y) en $(\mu x, \mu y)$.
Pour la construction de la nouvelle image, on créera une nouvelle liste de listes de dimensions $\lfloor \mu \mathrm{L} \rfloor$ lignes et $\lfloor \mu \mathrm{C} \rfloor$ colonnes. Un point en ligne I colonne J dont les coordonnées sont (X, Y) correspondra à un point (x, y) de l'image d'origine avec

$$x = \frac{1}{\mu}\mathrm{X} \qquad y = \frac{1}{\mu}\mathrm{Y}.$$

Pour la lisibilité de l'implémentation, on définira des fonctions locales `xy_ij` et `IJ_XY` qui réalisent les conversions ligne/colonne $\longleftrightarrow$ coordonnées.

Exercice 4

Écrire une fonction `rotation(tab,a)` qui réalise la rotation d'une l'image d'un angle a en transformant chaque point de coordonnées (x, y) en (X, Y) suivant la relation

$$\begin{pmatrix} \mathrm{X} \\ \mathrm{Y} \end{pmatrix} = \mathrm{R}(a) \begin{pmatrix} x \\ y \end{pmatrix} \quad \text{avec} \quad \mathrm{R}(a) = \begin{pmatrix} \cos(a) & -\sin(a) \\ \sin(a) & \cos(a) \end{pmatrix}.$$

Pour la construction de la nouvelle image, on considère qu'elle a les mêmes dimensions que l'image d'origine. Si un point sort du cadre après rotation, il est « perdu ». Un point de l'image pivotée en ligne I colonne J dont les coordonnées sont (X, Y) correspond à un point (x, y) de l'image d'origine avec

$$\begin{pmatrix} x \\ y \end{pmatrix} = \mathrm{R}(-a) \begin{pmatrix} \mathrm{X} \\ \mathrm{Y} \end{pmatrix}$$

autrement dit

$$\begin{cases} x = \cos(a)\mathrm{X} + \sin(a)\mathrm{Y} \\ y = -\sin(a)\mathrm{X} + \cos(a)\mathrm{Y} \end{cases}$$

Pour la lisibilité de l'implémentation, on définira des fonctions locales `xy_ij` et `ij_xy` qui réalisent les conversions ligne/colonne $\longleftrightarrow$ coordonnées. On définira également une fonction `deborde` qui détermine si un point ligne `i` colonne `j` déborde du cadre de l'image ou pas.