

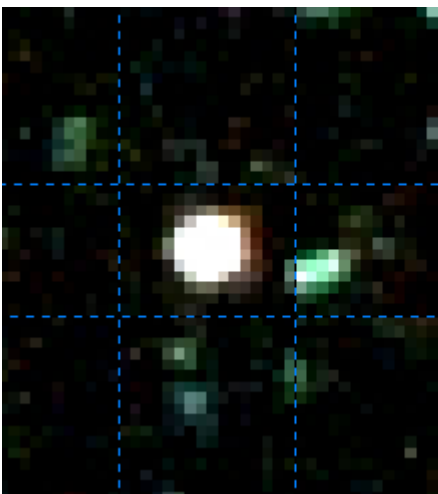
# ESTIMATION DE LA LUMINOSITÉ D'UNE ÉTOILE VARIABLE à partir d'une photographie.

Exercices préparatoires : opérations numériques sur **un** tableau de nombres

## 1. L'image de départ

L'image ci-contre est utilisée pour les exemples de traitements numériques qui suivent.

Dans cette image, une zone définie par ses coordonnées va être systématiquement explorée. (voir ci-dessous)



La zone concernée est limitée par le rectangle en pointillés. En utilisant la syntaxe du langage Python la zone est définie par : ligne du dessus, ligne du dessous, colonne de gauche, colonne de droite.

Ce qui amène aux variables suivantes :  
`li1,li2,co1,co2 = 74,82,118,129`

## 2. Bibliothèques et chargement de l'image en gamme de gris

```
from PIL import Image
from pylab import *
im = array(Image.open("TCas_a_cliquer2.jpg").convert("L"))
```

<code>im</code>	Nom symbolique à utiliser par la suite pour désigner l'image chargée dans la mémoire
<code>array</code>	Transforme en tableau à deux dimensions (hauteur, largeur)
<code>Image.open</code>	Tourne le robinet du "tonneau" qui contient l'image
<code>convert("L")</code>	Transforme l'image couleur (3 tableaux de données) en gamme de gris (un seul tableau).

## 3. Lister les valeurs des pixels de la zone `li1,li2,co1,co2`

On utilise la fonction `print` : `print(im[li1:li2,co1:co2])`

```
[[ 1  27  41  65 101 122  80  71  47  36  0]
 [ 1  21  91 193 255 255 246 173  84  36  1]]
```

```
[ 2  44 239 254 254 255 254 255 126  28  8 ]
[ 17 120 255 255 255 254 254 254 141  42  2 ]
[ 27 110 254 255 255 255 255 255 167  61  6 ]
[  0  30 210 254 255 255 255 253 114  44 24 ]
[  0  19  59 204 255 255 210 101  82  55  6 ]
[ 16  19  17  68 102  95  50  51  45  18  0 ]]
```

*Remarques :*

Chacun des pixels a une intensité définie par un nombre allant de 0 à 255.

Plus le nombre est gros, plus l'intensité est forte.

Comparer avec la zone définie par les pointillés ci-dessus.

#### **4. Créer un nouveau tableau par filtrage.**

Le filtrage remplace les intensités fortes par des zéros et les intensités faibles par des 1.

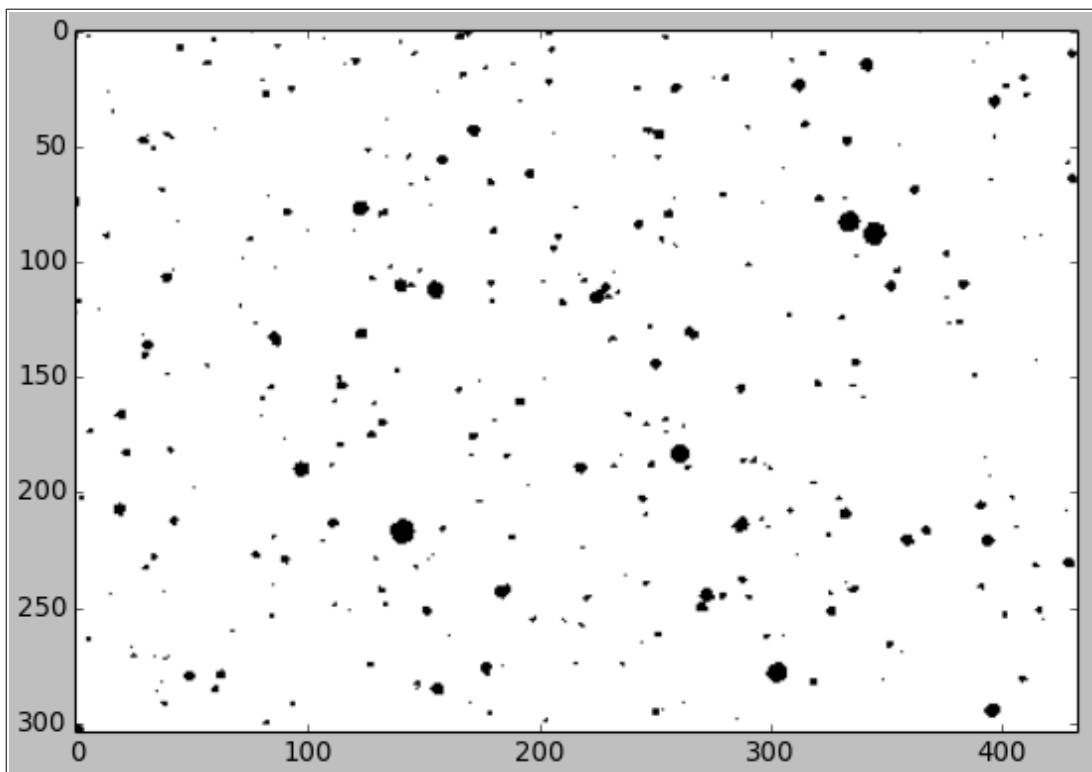
$image = 1*(im < s)$

En listant la zone définie par li1,li2,co1,co2 on obtient

```
[ [ 1 1 1 1 1 1 1 1 1 1 1 ]
  [ 1 1 1 0 0 0 0 0 1 1 1 ]
  [ 1 1 0 0 0 0 0 0 1 1 1 ]
  [ 1 1 0 0 0 0 0 0 0 1 1 ]
  [ 1 1 0 0 0 0 0 0 0 1 1 ]
  [ 1 1 0 0 0 0 0 0 1 1 1 ]
  [ 1 1 1 0 0 0 0 1 1 1 1 ]
  [ 1 1 1 1 1 1 1 1 1 1 1 ] ]
```

Comparer avec le listage précédent : on retrouve la forme de l'étoile.

Si l'on affiche l'image qui résulte du filtrage, on obtient :



## 5. Créer un autre tableau par le filtrage inverse

Le filtrage remplace les intensités fortes par des 1 et les intensités faibles par des 0.

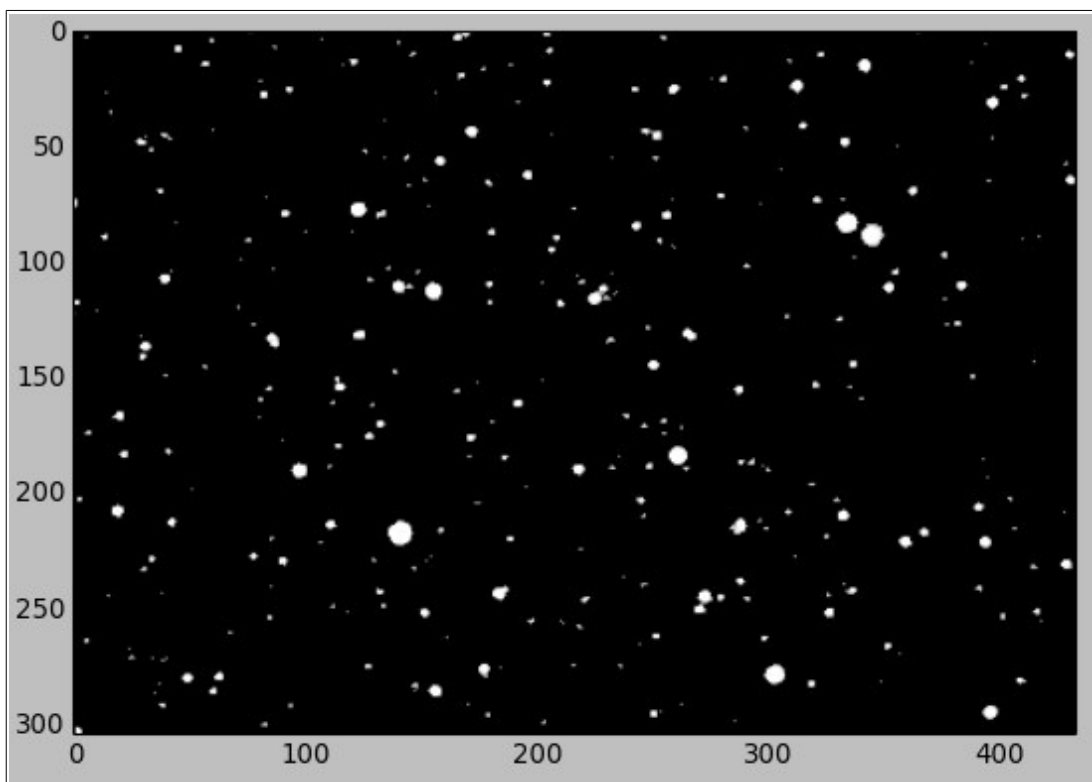
$$msq = 1*(im>s)$$

En listant la zone définie par li1,li2,co1,co2 on obtient :

```
[ [ 0 0 0 0 0 0 0 0 0 0 0 ]  
  [ 0 0 0 1 1 1 1 1 0 0 0 ]  
  [ 0 0 1 1 1 1 1 1 0 0 0 ]  
  [ 0 0 1 1 1 1 1 1 1 0 0 ]  
  [ 0 0 1 1 1 1 1 1 1 0 0 ]  
  [ 0 0 1 1 1 1 1 1 0 0 0 ]  
  [ 0 0 0 1 1 1 1 0 0 0 0 ]  
  [ 0 0 0 0 0 0 0 0 0 0 0 ] ]
```

Comparer avec les listages précédents : on retrouve la forme de l'étoile.

Si l'on affiche l'image qui résulte du filtrage, on obtient :



Ce dernier tableau de nombres va être utilisé comme masque de l'image initiale : à chaque fois qu'il contient un zéro le pixel correspondant de l'image sera mis à 0 (c'est à dire qu'il s'affichera en noir).

A chaque fois qu'il contient un 1 le pixel correspondant de l'image sera conservé à sa valeur.

Ce petit miracle se réalise par une multiplication de l'image par le masque.

Pour créer le nouveau tableau par cette opération, on écrit simplement :

$$im2 = im * msq$$

En listant la zone définie par li1,li2,co1,co2 on obtient :

(voir page suivante).

```
[ [ 0 0 0 0 0 0 0 0 0 0 0 0 ]
  [ 0 0 0 193 255 255 246 173 0 0 0 ]
  [ 0 0 239 254 254 255 254 255 0 0 0 ]
  [ 0 0 255 255 255 254 254 254 141 0 0 ]
  [ 0 0 254 255 255 255 255 255 167 0 0 ]
  [ 0 0 210 254 255 255 255 253 0 0 0 ]
  [ 0 0 0 204 255 255 210 0 0 0 0 ]
  [ 0 0 0 0 0 0 0 0 0 0 0 ] ]
```

Comparer avec les listages précédents : on retrouve la forme de l'étoile **et** les valeurs les plus faibles ont été éliminées.

Si l'on affiche l'image qui résulte du filtrage, elle ressemble à celle du masque.

## **6. Résumé**

Dans ce petit support, nous avons vu :

Comment charger une image JPEG et la convertir en gamme de gris.

Comment lister les valeurs des pixels sur une zone prédéfinie.

Comment effectuer le filtrage à partir d'un seuil pour réaliser un masque.

Comment appliquer un masque sur l'image de façon à ne garder que les pixels les plus "forts".

Il faudrait maintenant refaire les mêmes manipulations sur une image en couleur.