

chsh_images

November 26, 2021

1 Images et matrices

Quelques manipulations d'images avec deux modules [Pillow](#) et [scikit-image](#). Le premier module implémente les entrées sorties et quelques effet spéciaux, le second est pratique quand il faut travailler numériquement avec les images.

```
[1]: from jyquickhelper import add_notebook_menu
      add_notebook_menu()
```

[1]: <IPython.core.display.HTML object>

1.1 PIL : operations simples

1.1.1 Open

```
[2]: from PIL import Image
      img = Image.open("images1.jpg")
      img
```

[2]:



```
[3]: img.size
```

[3]: (289, 175)

```
[4]: img.resize((50, 50))
```

[4]:



1.1.2 Combiner

```
[5]: new_img = Image.new('RGB', (img.size[0]*2, img.size[1]))
new_img.paste(img, (0,0))
new_img.paste(img, (img.size[0],0))
new_img
```

[5]:



```
[6]: def combine(*imgs, mode='RGB', vert=False):
    if vert:
        sizesx = [im.size[0] for im in imgs]
        sizesy = [im.size[1] for im in imgs]
        new_img = Image.new(mode, (max(sizesx), sum(sizesy)))
        y = 0
        for im in imgs:
            new_img.paste(im, (0, y))
            y += im.size[1]
    else:
        sizesx = [im.size[0] for im in imgs]
        sizesy = [im.size[1] for im in imgs]
        new_img = Image.new(mode, (sum(sizesx), max(sizesy)))
        x = 0
        for im in imgs:
            new_img.paste(im, (x, 0))
            x += im.size[0]
    return new_img

combine(img, img)
```

[6]:



```
[7]: combine(img, img, vert=True)
```

[7]:



1.2 PIL to array

Une image en couleur contient trois images, une pour chaque couleur primaire.

```
[8]: import numpy
array = numpy.array(img.getdata(), dtype=numpy.uint8).reshape(img.size[1], img.
↪size[0], 3)
array.shape
```

[8]: (175, 289, 3)

```
[9]: array.dtype
```

[9]: dtype('uint8')

1.3 D'une matrice à sa transposée

```
[10]: array.transpose((2, 1, 0)).shape
```

[10]: (3, 289, 175)

1.4 Matrice à PIL

```
[11]: from PIL import Image
img2 = Image.fromarray(array)
img2
```

[11]:



1.5 Séparer les couleurs

```
[12]: im_r, im_b, im_g = img.split()
combine(im_r, im_b, im_g, mode="L")
```

[12]:



1.5.1 YCbCr

```
[13]: img_ycbcr = img.convert('YCbCr')  
img_ycbcr.size
```

[13]: (289, 175)

```
[14]: img_y, img_cb, img_cr = img_ycbcr.split()  
img_y.size
```

[14]: (289, 175)

```
[15]: combine(img_y, img_cb, img_cr, mode="L")
```

[15]:



[16]:

