

# 1 Python correction

## 1.1 imports

```
1 import Queue
from scipy import misc
3 import matplotlib.pyplot as plt
import numpy as np
```

## 1.2 Predicate

This function defines the agregation condition.

```
def predicate(image, i, j, seed) :
2     f=image[i,j];
    g=image[seed[0], seed[1]];
4     return abs(f-g)<20
```

The following code is used to start the region growing from a pixel manually clicked on an image.

```
# start of code
2 fig = plt.figure();
ax = fig.add_subplot(211);
4 ax.set_title('Click on a point')

6 # load lena image
lena = misc.lena();
8 ax.imshow(lena, picker=True, cmap=plt.gray());

10 fig.canvas.mpl_connect('button_press_event', onpick)
plt.show();
```

And here comes the main function for region growing.

```

1 def onpick(event):
2     #print 'x=%d, y=%d, xdata=%f, ydata=%f'%(
3     #     event.x, event.y, event.xdata, event.ydata)
4     # pixel d'origine
5     seed = np.array([int(event.ydata), int(event.xdata)]);
6
7     queue = Queue.Queue();
8     queue.put(seed);
9
10    # Visited matrix : result of segmentation
11    # this matrix will contain 1 if in the region,
12    #                                     -1 if visited but not in the region
13    #                                     0 if not visited
14    visited = np.zeros(lena.shape)
15
16    #**** Start of algorithm ****
17    visited[seed[0], seed[1]] = 1;
18
19    while not queue.empty():
20        p = queue.get();
21
22        for i in range(max(0,p[0]-1), min(lena.shape[0],p[0]+2)):
23            for j in range(max(0,p[1]-1), min(lena.shape[1],p[1]+2)):
24                if not visited[i,j]:
25                    if predicate(lena, i, j, seed) :
26                        visited[i,j] = 1;
27                        queue.put(np.array([i,j]));
28                    else :
29                        visited[i,j] = -1;
30
31    #*****
32    # end of the algorithm:
33    # the visited matrix contains the segmentation result
34
35    # display results
36    ax=fig.add_subplot(212);
37    ax.imshow(visited==1);
38    fig.canvas.draw();

```