1 Python correction

```
from skimage.util import random_noise
from scipy import misc
import matplotlib.pyplot as plt
from skimage import morphology as m
from skimage import filters
import numpy as np
```

1.1 Morphological center

The noisy image is obtained with the function random_noise from skimage.util.

```
L = misc.imread('lena512.bmp');
A = random_noise(L, mode='s&p', amount=.04);
```

Following the definition, the morphological center is obtained with the following code, and illustrated in Fig.1:

```
def morphoCenter(I, c, o, selem=m.disk(1)):
    """
    """
    coc = c(o(c(I, selem=selem), selem=selem), selem=selem);
    oco = o(c(o(I, selem=selem), selem=selem), selem=selem);
    cMin = np.minimum(oco, coc);
    cMax = np.maximum(oco, coc);
    F = np.minimum( np.maximum(A, cMin), cMax);
    return F;

B = morphoCenter(A, m.closing, m.opening);
Bmed= filters.median(A, selem=np.ones((3,3)));
```

1.2 Alternate sequential filters

The order of these filters are often chosen empirically. The results are illustrated in Fig.2.

(a) Noisy image.

(b) Median filter of size 5×5.

(c) Morphological center.

Figure 1: Morphological center compared to the classical median filter.

```
def asf_n(I, order=3):
    F = I.copy();
    for r in np.arange(1, order+1):
        se = m.disk(r);
        F = m.opening( m.closing(F, selem=se), selem=se);
    return F;
```

```
def asf_m(I, order=3):
    F = I.copy();
    for r in np.arange(1, order+1):
        se = m.disk(r);
        F = m.closing( m.opening(F, selem=se), selem=se);
    return F;
```

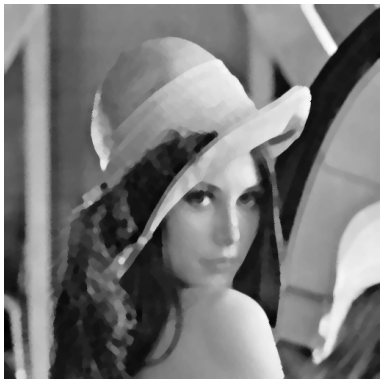
1.3 Geodesic reconstruction filters

These two functions are simply implemented using erosion and reconstruction operators. Notice the duality property, that is used to code closerec. In this example, 8 bits images (unsigned) are considered, and the results are illustrated in Fig.3.

```
def openrec(I, selem=m.disk(1)):
    B = m.erosion(I, selem=selem);
    F = m.reconstruction(B, I);
    return F;
```

(a) Original image.

(b) ASF of order 3, starting with a closing operation (denoted N).



(c) ASF of order 3, starting with an opening operation (denoted M).

Figure 2: Alternate Sequential Filters compared to original image.

```
def closerec(I, selem=sm.disk(1)):
    F = 255-openrec(255-I, selem=selem);
    return F;
```

(a) Original image.

(b) Opening by reconstruction.

(c) Closing by reconstruction.

Figure 3: Opening and closing by reconstruction.

1.4 ASF by reconstruction

This is an example of a 3rd order alternate sequential filter, illustrated in Fig.4.

```
def asfrec(I, order=3):
    A = I.copy();
    for r in np.arange(1, order+1):
        se = m.disk(r);
        A = closerec(openrec(A, selem=se), selem=se);
    return A;
```

(a) Original image.

(b) ASF of order 3 by reconstruction.

(c) Noisy image.

Figure 4: Alternate Sequential Filtering by reconstruction.

1.5 Morphological center by reconstruction

Morphological center by reconstruction replaces the opening and closing operations by their equivalent by reconstruction (see Fig.5).

```
def centerrec(I, selem=m.disk(1)):
    """
    """
    B = morphoCenter(I, closerec, openrec, selem=selem);
    return B;
```

(a) Original image. (b) Center by reconstruction. (c) Noisy image.

Figure 5: Morphological center by reconstruction.