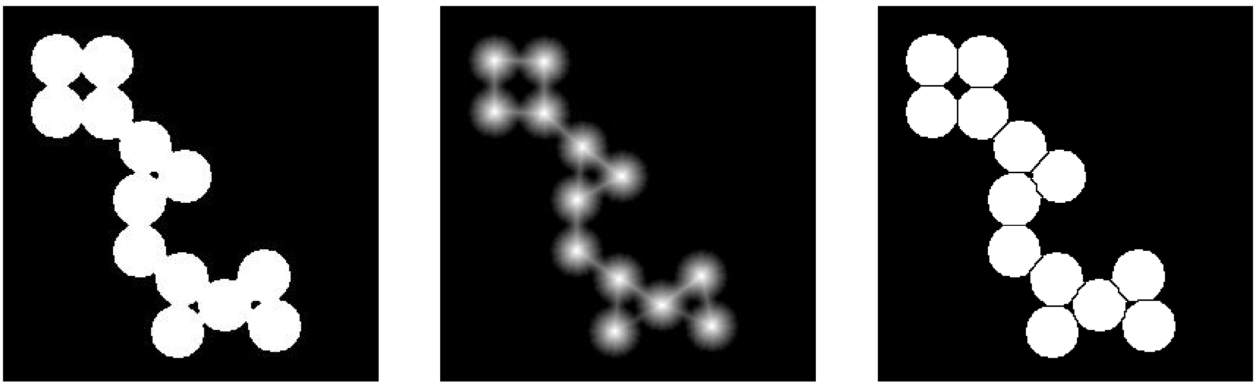
1 Matlab correction

1.1 Watershed and distance map

This method is a classical way of performing the separation of some objects by proximity or influence zones. It is illustrated in Fig.1.

```
A=imread('circles.tif');
% distance map
dist=bwdist(~A);
% watershed
watf=watershed(imcomplement(dist));
watf=(watf==0);
% separation of the grains
B=A & ~watf;
```



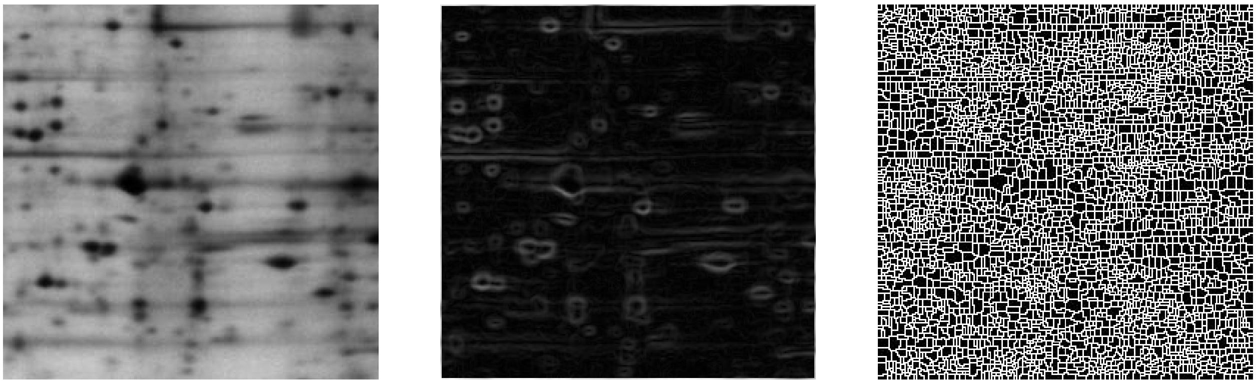
(a) Original image. (b) Distance map. (c) Separation of the grains.

Figure 1: Steps of the separation of the grains.

1.2 Watershed and image gradients

The gradient image amplifies the noise. Thus, the watershed operator directly applied to the gradient of the image produces an over-segmented image (see Fig.1).

```
% read grayscale image
A=imread('gel.jpg');
% gradient
gradient=sobel(A);
rm=imregionalmin(gradient);
% watershed
wat=watershed(gradient);
wat=(wat==0);
```



(a) Original image. (b) Amplitude of the gradient (Sobel). (c) Watershed segmentation.

Figure 2: Performing the watershed on the gradient image is not a good idea.

In fact, this method produces as many segments as there are minima in the gradient image Fig.3.

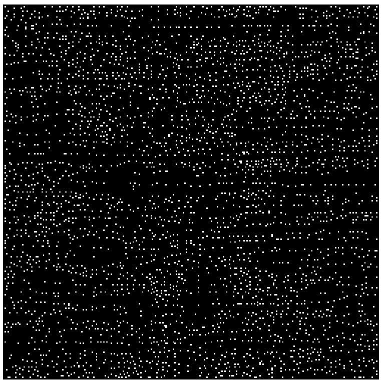


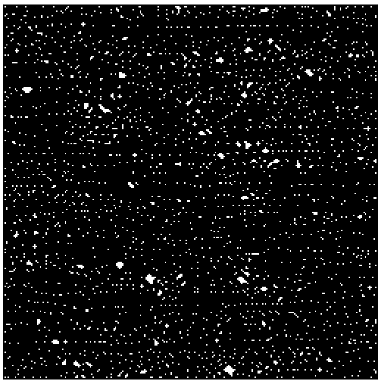
Figure 3: Local minima of the gradient image.

1.2.1 Solution: filtering the image

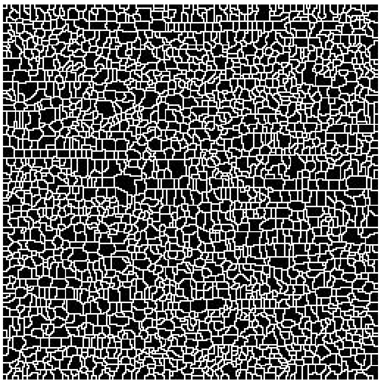
Before evaluating the gradient, the image is filtered. The number of minima is lower and this leads to a less over-segmented image (Fig.4).

```
A=imread('gel.jpg');
% filtering
se = strel('disk',2);
AA=imopen(A,se);
f=imclose(AA,se);
% gradient
gradient=imgradient(f);
rm=imregionalmin(gradient);
% watershed
```

```
wat=watershed(gradient);
wat=(wat==0);
```



(a) Minima of the gradient after filtering the image.



(b) Watershed segmentation.

Figure 4: Even if the gradient is performed on the filtered image, there is still a high over-segmentation.

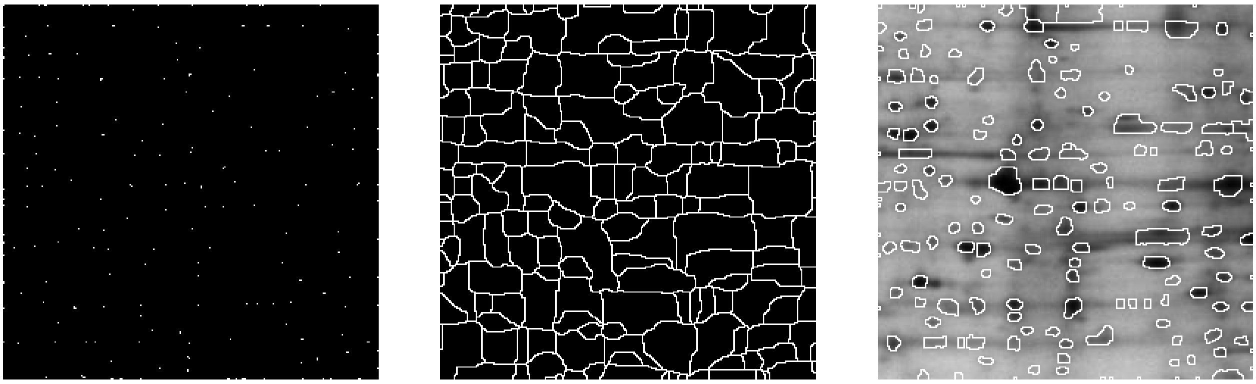
1.3 Watershed constrained by markers

The watershed can be constrained by markers: the markers can provide the correct number of regions. This method imposes both the background (external markers) and the objects (internal markers). The results are illustrated in Fig.5. The ultimate erosion of the internal markers is used to deconnect these markers from the external markers.

```
% filtering
se = strel('disk',2);
AA=imopen(A,se);
f=imclose(AA,se);
% internal markers (of the objects)
rm=imregionalmin(f);
rm = bwulterode(rm);
% external markers: watershed of filtered image
% background of the objects
watf=watershed(f);
watf=(watf==0);

% contrain the minima
gradient=imgradient(f);
mie=imimposemin(gradient, rm | watf);
minima=max(rm,watf);
% watershed constraint
watc=watershed(mie);
```

```
watc=(watc==0);
```



(a) Internal markers (blobs). (b) External (background) markers . (c) Final segmentation.

Figure 5: Watershed segmentation by markers.